

ANALISIS DAN IMPLEMENTASI ALGORITMA LRU DAN GDSF SEBAGAI ATURAN CHASE REPLACEMENT PADA *PROXY SERVER*

Gunadi Pratama¹, Wawan Kusdiawan² & Jajang Mulyana³

^{1,2,3} Informatika, STMIK Kharisma Karawang Jl. Pangkal Perjuangan Km 1, Karawang 41361, Indonesia
Email: ja2ngm@gmail.com

Abstrak. Salah fungsi *proxy server* pada jaringan komputer adalah *caching*, dimana mekanisme *caching* pada *proxy server* adalah dengan menyimpan objek-objek yang merupakan hasil permintaan komputer *client* dari internet dan memberikan layanan jika *client* akan mengaksesnya kembali tanpa meminta sepenuhnya ke internet. Sehingga hal ini akan menyebabkan *cache memory* menjadi penuh. Untuk itu salah satu solusinya adalah dengan cara mengoptimalkan kinerja *cache memory* pada *proxy server caching* yang terdapat metode *cache replacement* pada *proxy server*. *Cache replacement* pada *proxy server* merupakan metode penghapusan objek pada *cache memory* untuk digantikan dengan objek baru yang bertujuan agar *cache memory* tidak penuh. Maka dari itu penulis akan mengimplementasikan hasil analisis algoritme *Least recently used* dan *Greedy dual size frequently* sebagai aturan *cache replacement* pada *squid proxy* dengan menggunakan metode pengembangan *Network Development Live Cycle* (NDLC) Cisco PPDIOO. Hasil yang dicapai dalam penelitian ini berupa data informasi hasil perbandingan algoritme *cache replacement* pada *proxy server caching*. Penggunaan *proxy server* sebagai web cache dapat diterapkan pada sebuah instansi atau lembaga yang sering mengakses web http secara intensif seperti lembaga pemerintahan dan pendidikan.

Kata kunci: *Proxy server, Caching, Cache Replacement, LRU, GDSF, Cisco PPDIOO.*

1 Pendahuluan

Hasil survei Asosiasi Penyelenggara Jasa Internet Indonesia (APJII) pada tahun 2017, pengguna internet di Indonesia mencapai 143,26 Juta jiwa dari total populasi penduduk Indonesia sebanyak 262 Juta jiwa atau mencapai 54,68%^[1]. Dengan banyaknya pengguna internet maka diperlukan bandwidth internet yang memadai serta manajemen pengelolaan bandwidth yang baik agar kegiatan informasi dan komunikasi dapat berjalan dengan baik. Dalam sebuah jaringan komputer terdapat *proxy server* yang memiliki fungsi salah satunya yaitu *caching* yang dapat mengoptimalkan bandwidth internet, mekanisme *caching* pada *proxy server* yaitu menyimpan data-data permintaan akses dari komputer-komputer *client* yang ditujukan ke internet atau server tujuan kemudian memberikan layanan kepada *client* lain jika terjadi pengaksesan data yang sama tanpa meminta layanan ke internet atau server tujuan sepenuhnya.

Dengan adanya fungsi *caching* pada *proxy server* maka bandwidth internet tidak akan terpakai sepenuhnya. Karena *cache memory* memiliki ukuran yang terbatas dan *proxy server* harus menyediakan ruang yang cukup oleh karena itu *cache replacement* sangat dibutuhkan. Tujuan dari aturan *cache replacement* ini adalah untuk memilih objek yang terakhir kali digunakan, objek dengan ukuran yang besar serta objek mana yang lebih sering digunakan oleh *client* yang mengakses pada *Proxy server* ^[2].

Dalam hal ini penulis telah mendapatkan referensi dari penelitian sebelumnya yang dilakukan Yuisar, 2015 yang menghasilkan *load time* dan *avarage speed* yang lebih baik dari sebelum dilakukan *cache*, tetapi masih pada proses *cache replacement* yang terlambat. Penelitian yang relevan juga yang telah dilakukan oleh Fata Nidaul Khasanah, 2017 yang menghasilkan meningkatnya kecepatan akses internet sehingga permasalahan akses internet yang lambat dapat teratasi dikarenakan situs yang telah dibuka oleh client sebelumnya tersimpan pada jaringan local sehingga client selanjutnya yang membuka situs yang sama tidak perlu melakukan *request* ke internet. Penelitian yang relevan yang berkaitan dengan mekanisme *replacement* pada memory yang telah dilakukan oleh Abas Ali Pangera, 2013 tentang algoritme *page replacement* yang menjelaskan Algoritme LRU merupakan perpaduan dari algoritme FIFO dan optimal. Prinsip dari algoritme LRU adalah mengganti *page* yang sudah tidak digunakan untuk periode waktu terlama.

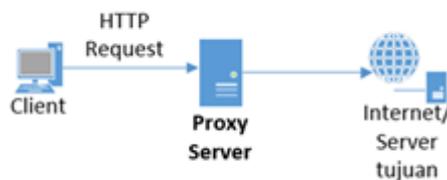
Dalam algoritme *cache replacement*, faktor utama yang perlu dipertimbangkan adalah fungsi yang mencakup *recency*, *frequency*, *cost*, *size*, *expiration time* dan *modification time* [3]. Berdasarkan latar belakang masalah dan referensi penelitian sebelumnya maka penulis akan melakukan penelitian mengenai analisis dan implementasi algoritme *Least Recently Used* (LRU) dan *Algoritme Greedy Dual Size Frequency* (GDSF) sebagai aturan *cache replacement* pada *proxy server* untuk mengoptimalkan kinerja *cache memory* pada *proxy server* dengan menggunakan metode penelitian *Netwok Development Live Cycle* (NDLC) Cisco *Plan, Prepare, Design, Implement, Operate and Optimize* (PPDIOO) [4]. Berdasarkan uraian diatas maka penulis mengambil judul penelitian “Analisis dan Implementasi Algoritme LRU dan GDSF Sebagai Aturan *Cache Replacement* Pada *Proxy server*”.

2 Metodologi Penelitian

2.1 Bahan Penelitian

2.1.1 Kebutuhan Organisasi

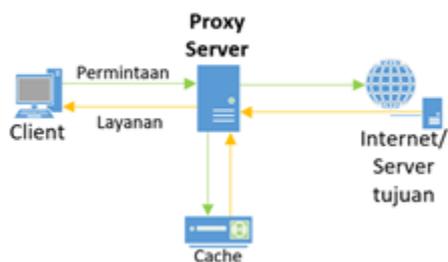
Kebutuhan proxy server pada sebuah organisasi jaringan adalah sebagai jembatan antara computer client dengan server tujuan (internet). Proxy server akan melayani permintaan koneksi client dan meneruskan kepada server tujuan



Gambar 1. Tata letak Proxy server

2.1.2 Konsep Strategi dan Arsitektur Proxy server

Pada penelitian ini proxy server akan diposisikan sebagai web caching dengan tujuan menyimpan data access log client dan memberikan kepada client lain ketika permintaannya sama tanpa mengambil data keseluruhan dari server tujuan.



Gambar 2. Proxy server caching

2.2 Alat Penelitian

Alat penelitian yang digunakan pada penelitian ini berupa perangkat keras (*hardware*) yaitu komputer yang akan bertindak sebagai *server* dan *client* dan perangkat lunak (*software*) berupa *daemon squid proxy*.

2.2.1 Kebutuhan *Hardware*

Hardware merupakan komponen penting dalam kegiatan analisis dan implementasi yang dilakukan oleh penulis. Perangkat keras (*hardware*) yang dibutuhkan yaitu :

Tabel 1. Kebutuhan *Hardware*

No.	Perangkat	Keterangan
1.	Komputer Server	Spesifikasi komputer server menggunakan processor Core i7 8700 3,4 GHz dengan jumlah core 12, RAM 8GB dan hardisk 1 TB.
2.	Komputer <i>Client</i>	menggunakan <i>processor</i> minimal <i>Dual Core</i> , memiliki <i>port</i> koneksi LAN dan memiliki aplikasi browser.
3.	Media Penghubung	Media penghubung antara <i>client</i> dan server menggunakan media kabel (<i>wire network</i>).

Komputer Server sebagai perangkat jaringan yang memberikan layanan terhadap pengguna. Server melayani pengguna dengan handal dan harus bisa melakukannya setiap saat [5], Personal komputer untuk keperluan konfigurasi server, sebagai media untuk pengujian analisis, *Switch* sebagai alat untuk penghubung antar komputer, Kabel UTP media transfer data atau Access Point sebagai opsional media transfer data.

2.2.2 Kebutuhan *Software*

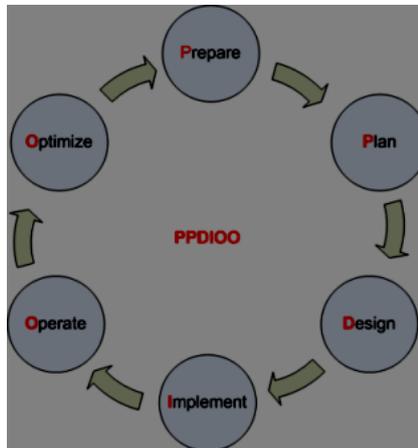
Software yang digunakan dalam penelitian ini diantaranya sebagai berikut :

Tabel 2. Kebutuhan *Software*

No.	Perangkat Lunak	Keterangan
1.	Sistem operasi Ubuntu Server	Pada penelitian ini ubuntu server digunakan sebagai sistem operasi server yang dapat menjalankan <i>squid proxy</i> dengan versi ubuntu server 14.04 LTS.
2.	<i>Squid Proxy</i>	<i>Squid</i> merupakan <i>daemon</i> yang akan digunakan sebagai <i>proxy server</i> dan web <i>cache</i> pada penelitian ini.
3.	Putty	Putty digunakan sebagai aplikasi <i>remote desktop</i> (SSH) yang digunakan untuk mengkonfigurasi server.
4.	<i>Squid analyzer</i>	<i>Squid analyzer</i> merupakan <i>daemon</i> dengan output PHP Base atau aplikasi web yang digunakan sebagai media monitoring pada Linux dan dapat dikonfigurasi pada <i>squid proxy</i> .

2.3 Metode Pengembangan Sistem

Metode yang digunakan untuk mengembangkan sistem ini adalah metode penelitian *Network Development Live Cycle* (NDLC) Cisco PPDIOO. Yang terdiri dari *Plan, Prepare, Design, Implement, Operate* dan *Optimize* ^[4].



Gambar 3. NDLC Cisco PPDIOO

Pada gambar 3 mendeskripsikan sebuah model siklus hidup jaringan dengan konsep PPDIOO yaitu, Prepare (persiapan), Plan (Perencanaan), Design (Desain), Implement (Implementasi), Operate (Operasi) dan Optimize (Optimasi).

3 Hasil dan Pembahasan

3.1 Kebutuhan Organisasi Media Penyimpanan Cache

Tabel 3. Kebutuhan media penyimpanan *cache*

No.	Kebutuhan	Keterangan
1.	Penyimpanan <i>cache</i> ganda (2x100 MB)	Penyimpanan <i>cache</i> ganda akan diterapkan dan akan digunakan keduanya.
2.	BTRFS	BTRFS atau <i>B-tree File System</i> adalah file sistem masa depan yang memiliki arsitektur penyimpanan seperti ranting pohon yang dapat mengatur mekanisme penyimpanan maupun pencarian data dari media penyimpanan, agar media penyimpanan semakin <i>reliable</i> , dapat diakses dalam waktu cepat. ^[6]

3.2 Analisis Algoritme *Cache Replacement*

Algoritme *cache replacement* yang digunakan dalam kegiatan analisis dan implementasi yang dilakukan oleh penulis antara lain adalah :

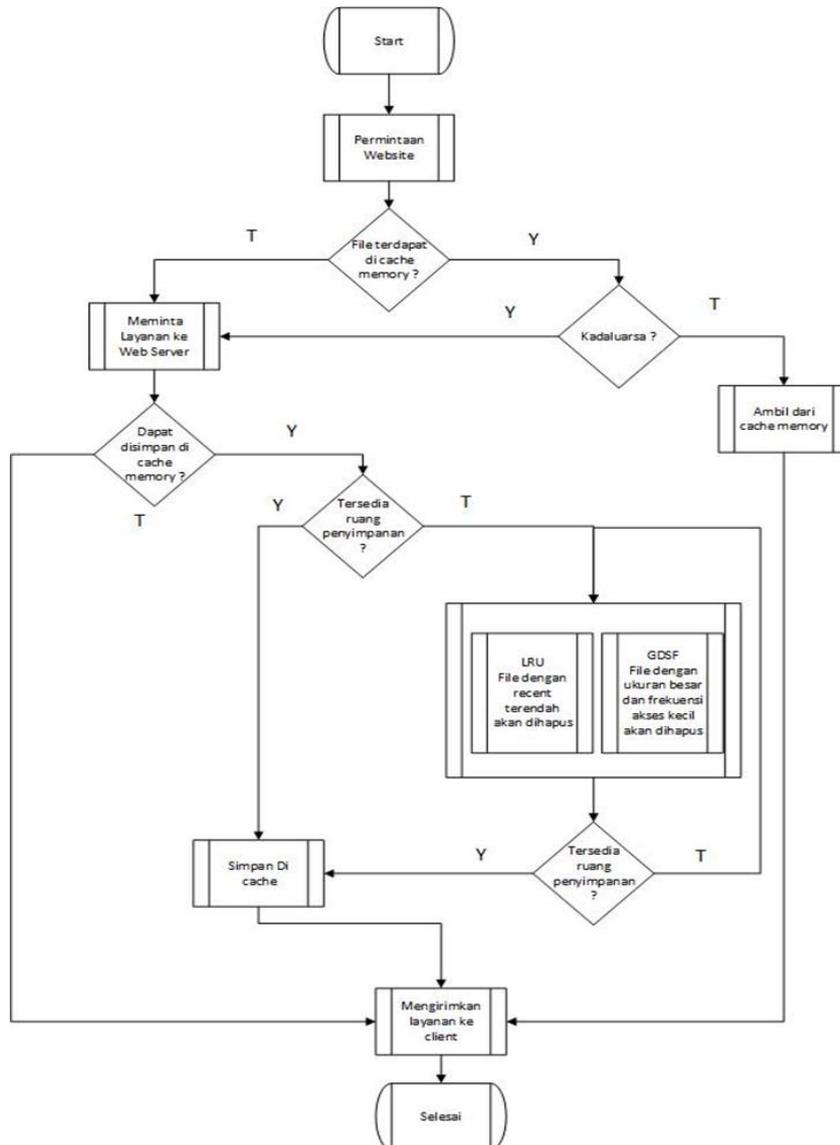
Tabel 4. Algoritme *Cache Replacement*

No.	Algoritme	Keterangan
1.	<i>Least recently used</i> (LRU)	LRU merupakan salah satu algoritme yang cukup baik dalam <i>page replacement</i> . LRU menganut sebuah prinsip yang cukup sederhana, yaitu membuang proses yang paling lama tersimpan pada memori atau paling jarang digunakan dari sekumpulan proses-proses yang ada. Karena algoritme ini mengasumsikan bahwa proses yang baru saja dijalankan kemungkinan besar akan dijalankan kembali dalam waktu dekat ^[7] .

2. *Greedy dual size frequently* (GDSF) Algoritme *Greedy dual size frequently* (GDSF) adalah dasar penggantian atau penghapusan data dengan melakukan perbandingan ukuran (*size*) dan seberapa sering akses pada file tersebut dilakukan (*frequency*). Setelah mengkaji besar ukuran file dan tingkat frekuensi pengaksesan data pada *cache memory*, selanjutnya akan dilakukan penghapusan sekaligus penggantian data baru yang akan disimpan pada *cache memory*.

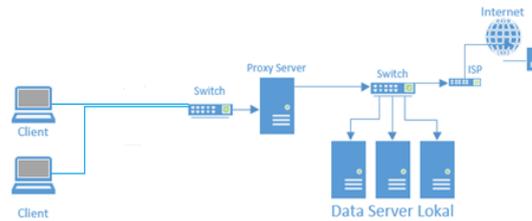
3.3 Flowchart

Fase *plan* merupakan tahapan identifikasi persyaratan jaringan berdasarkan tujuan, fasilitas dan kebutuhan pengguna, pada penelitian ini diantaranya adalah sebagai berikut :



Gambar 4. Flowchart proxy server

3.4 Topologi Jaringan



Gambar 5. Proxy server topologi

3.5 Instalasi sistem

3.5.1 Instalasi Linux Ubuntu Server

Pada instalasi ubuntu server yang digunakan sebagai sistem operasi untuk menjalankan proxy server perlu memperhatikan pembagian partisi harddisk yang akan digunakan seperti partisi untuk sistem (root), partisi untuk booting, swap dan partisi cache sebagai media yang digunakan oleh squid untuk menyimpan data-data dokumen web yang diakses oleh client. Untuk skema partisinya adalah sebagai berikut :

Tabel 5 . Parameter instalasi ubuntu server.

No.	Mount Point	Size	File system	Location	Type
1.	/	900 GB	Ext4	Begining	Primary
2.	Efiboot	1 GB	Fat32	Begining	Primary
3.	Swap	16 GB	Swap	Begining	Primary
4.	/cache1	100 MB	Btrfs	Begining	Logical
5.	/cache2	100 MB	Btrfs	Begining	Logical

3.5.2 Instalasi paket dependency Squid

Tabel 6. File dependency

No.	Nama Paket	Keterangan
1.	<i>Squidclient</i>	Sebagai <i>squid reporting</i>
2.	<i>Libfile-readbackwards-perl</i>	Modul <i>perl</i> untuk pembacaan file mundur berdasarkan baris
3.	<i>GCC</i>	<i>GNU C Compiler</i>
4.	<i>Build-essential</i>	Berisi paket untuk megkompilasi <i>source C/C++</i>
5.	<i>Acpid</i>	<i>Advanced configuration and power interface event daemon</i>
6.	<i>Ccze</i>	<i>Log colorizer</i> untuk file <i>access.log</i>
7.	<i>Squid proxy</i>	<i>Daemon Proxy server</i>

3.6 Parameter Pengujian Algoritme Cache Replacement

Pengujian setiap algoritma cache replacement menggunakan beberapa website dan dibagi menjadi tiga tahap diantaranya sebagai berikut :

Tabel 7. Parameter pengujian

No.	Tahapan	Bentuk Pengujian
1.	Tahap 1	1. Pengujian pada 45 website pemerintahan (.go.id)
		2. Pengujian pada 45 website Pendidikan (.ac.id)
		3. Pengujian pada 45 website umum (.com, .blogspot.com dll.)
2.	Tahap 2	1. Pengujian pada 45 website umum (.com, .blogspot.com dll.)
		2. Pengujian pada 45 website Pendidikan (.ac.id)

- | | | |
|----|---------|--|
| 3. | Tahap 3 | <ol style="list-style-type: none"> 3. Pengujian pada 45 website pemerintahan (.go.id) 1. Pengujian pada 45 website umum (.com, .blogspot.com dll.) 2. Pengujian pada 45 website pemerintahan (.go.id) 3. Pengujian pada 45 website Pendidikan (.ac.id) |
|----|---------|--|
-

3.7 Pengambilan Data Perbandingan Algoritme

Pengambilan data perbandingan algoritme cache replacement menggunakan metode dari squidclient dan dengan menggunakan squidanalyzer dengan acuan hit ratio dan byte hit ratio dari informasi yang diberikan squid melalui file access.log. Pada informasi yang diberikan access.log, dimana jika permintaan client tidak dapat dipenuhi oleh cache maka hasil outputnya TCP_MISS, UDP_MISS dll. dan jika permintaan dapat dipenuhi oleh cache maka hasil outputnya TCP_HIT, TCP_REFRESH_HIT dll [8]. Hasil dari *access.log* akan di kalkulasi dan menghasilkan data *hit ratio* dan *byte hit ratio*.

3.8 Pengambilan Data Pengujian Dengan Squidclient

Pengambilan data menggunakan *squidclient* memiliki *output* dengan *type text base* dan bisa di akses langsung melalui perintah `$squidclient mgr:info / more` :

```

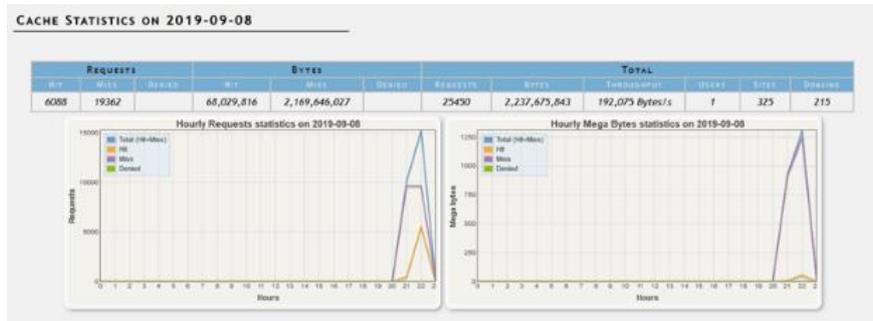
root@gunadi: /var/www/html/gunadi
root@gunadi:/var/www/html/gunadi# squidclient mgr:info | more
HTTP/1.0 200 OK
Date: Sun, 08 Sep 2019 15:56:31 GMT
Content-Type: text/plain
Expires: Sun, 08 Sep 2019 15:56:31 GMT
X-Cache: MISS from proxy.gunai.p.ht
X-Cache-Lookup: MISS from proxy.gunai.p.ht:3128
Connection: close

Squid Object Cache: Version LUSCA HEAD-r14809
Start Time: Sun, 08 Sep 2019 14:26:57 GMT
Current Time: Sun, 08 Sep 2019 15:56:31 GMT
Connection information for squid:
  Number of clients accessing cache: 2
  Number of HTTP requests received: 25636
  Number of ICP messages received: 0
  Number of ICP messages sent: 0
  Number of queued ICP replies: 0
  Number of HTCP messages received: 0
  Number of HTCP messages sent: 0
  Request failure ratio: 0.00
  Average HTTP requests per minute since start: 286.2
  Average ICP messages per minute since start: 0.0
  Select loop called: 2865520 times, 1.875 ms avg
Cache information for squid:
  Request Hit Ratios: 5min: 0.0%, 60min: 35.6%
  Byte Hit Ratios: 5min: -0.0%, 60min: 4.6%
  Request Memory Hit Ratios: 5min: 0.0%, 60min: 20.9%
  Request Disk Hit Ratios: 5min: 0.0%, 60min: 72.3%
  Storage Swap size: 111884 KB
  Storage Mem size: 8188 KB
  Mean Object Size: 18.47 KB
  Requests given to unlinkd: 0
    
```

Gambar 6. Squid reporting (Squidclient)

3.9 Pengambilan Data Pengujian Dengan Squidanalyzer

Pengambilan data menggunakan squidanalyzer dilakukan pada browser dan memiliki output php base dengan menampilkan hasil hit ratio dan byte hit ratio dalam bentuk tabel serta dilengkapi dengan grafik :



Gambar 7. Squid monitoring (Squidanalyzer)

3.9.1 Hasil pengujian Algoritme LRU dan GDSF (Squidanalyzer)

a. Hasil Algoritme LRU

Tabel 8. Hasil pengujian Algoritme LRU

No.	Hasil Pengujian	Jumlah
1.	Jumlah permintaan http	25110
2.	Jumlah hit	0.8% (200 hit)
3.	Jumlah miss	99.2% (24909 miss)
4.	Jumlah Permintaan Byte	-
5.	Jumlah byte hit	0.6%
6.	Jumlah byte miss	99.4 %

Dari tabel hasil pengujian algoritme *Least recently used* (LRU) dengan *squidclient* diperoleh data permintaan http sebanyak 25.110 dengan jumlah *hit* sebesar 0.8% atau 200 *hit* dengan perbandingan jumlah *miss* 99.2% atau 24.909. *squidclient* tidak menampilkan informasi mengenai jumlah permintaan *byte* tetapi hanya memberikan informasi *byte hit* sebesar 0.6% dan *byte miss* sebesar 99.4%.

```

root@gunadi: ~
HTTP/1.0 200 OK
Date: Sun, 08 Sep 2019 06:05:44 GMT
Content-Type: text/plain
Expires: Sun, 08 Sep 2019 06:05:44 GMT
X-Cache: MISS from proxy.gunai.p.ht
X-Cache-Lookup: MISS from proxy.gunai.p.ht:3128
Connection: close

Squid Object Cache: Version LUSCA_HEAD-rl4809
Start Time: Sun, 08 Sep 2019 04:54:10 GMT
Current Time: Sun, 08 Sep 2019 06:05:44 GMT
Connection information for squid:
  Number of clients accessing cache: 2
  Number of HTTP requests received: 25110
  Number of ICP messages received: 0
  Number of ICP messages sent: 0
  Number of queued ICP replies: 0
  Number of HTCP messages received: 0
  Number of HTCP messages sent: 0
  Request failure ratio: 0.00
  Average HTTP requests per minute since start: 350.8
  Average ICP messages per minute since start: 0.0
  Select loop called: 2869787 times, 1.496 ms avg
Cache information for squid:
  Request Hit Ratios: 5min: 0.1%, 60min: 0.8%
  Byte Hit Ratios: 5min: 0.1%, 60min: 0.6%
  Request Memory Hit Ratios: 5min: 0.0%, 60min: 46.4%
  Request Disk Hit Ratios: 5min: 100.0%, 60min: 46.7%
  Storage Swap size: 121776 KB
  Storage Mem size: 2924 KB
  Mean Object Size: 89.61 KB
  Requests given to unlink: 0
    
```

Gambar 8. Hasil pengujian Algoritme LRU (Squidclient)

b. Hasil Algoritme GDSF

Tabel 9. Hasil pengujian algoritme GDSF

No.	Hasil Pengujian	Jumlah
1.	Jumlah permintaan http	25.636
2.	Jumlah <i>hit</i>	35.6% (9.126 <i>hit</i>)
3.	Jumlah <i>miss</i>	64,4% (16.509 <i>miss</i>)
4.	Jumlah Permintaan <i>Byte</i>	-
5.	Jumlah <i>byte hit</i>	4.6%
6.	Jumlah <i>byte miss</i>	95.4%

Dari tabel hasil pengujian algoritme *greedy dual size frequently* (GDSF) dengan *squidclient* diperoleh data permintaan http sebanyak 25.636 dengan jumlah *hit ratio* sebesar 35.6% atau 9.126 *hit* dengan perbandingan jumlah *miss* 64.4% atau 16.509 *miss*. *squidclient* tidak menampilkan informasi mengenai jumlah permintaan *byte* tetapi hanya meberikan informasi *byte hit ratio* sebesar 4.6% dan *byte miss* sebesar 95.4%.

```

root@gunadi: /var/www/html/gunadi
root@gunadi:/var/www/html/gunadi# squidclient mgr:info | more
HTTP/1.0 200 OK
Date: Sun, 08 Sep 2019 15:56:31 GMT
Content-Type: text/plain
Expires: Sun, 08 Sep 2019 15:56:31 GMT
X-Cache: MISS from proxy.gunai.p.ht
X-Cache-Lookup: MISS from proxy.gunai.p.ht:3128
Connection: close

Squid Object Cache: Version LUSCA HEAD-r14809
Start Time: Sun, 08 Sep 2019 14:26:57 GMT
Current Time: Sun, 08 Sep 2019 15:56:31 GMT
Connection information for squid:
  Number of clients accessing cache: 2
  Number of HTTP requests received: 25636
  Number of ICP messages received: 0
  Number of ICP messages sent: 0
  Number of queued ICP replies: 0
  Number of HTCP messages received: 0
  Number of HTCP messages sent: 0
  Request failure ratio: 0.00
  Average HTTP requests per minute since start: 286.2
  Average ICP messages per minute since start: 0.0
  Select loop called: 2865520 times, 1.875 ms avg
Cache information for squid:
  Request Hit Ratios: 5min: 0.0%, 60min: 35.6%
  Byte Hit Ratios: 5min: -0.0%, 60min: 4.6%
  Request Memory Hit Ratios: 5min: 0.0%, 60min: 20.9%
  Request Disk Hit Ratios: 5min: 0.0%, 60min: 72.3%
  Storage Swap size: 111884 KB
  Storage Mem size: 8188 KB
  Mean Object Size: 18.47 KB
  Requests given to unlinkd: 0
    
```

Gambar 9. Hasil pengujian algoritme GDSF (Squidclient)

Tabel 10. Hasil perbandingan algoritme LRU & GDSF(Squidclient)

Algoritme	<i>Hit ratio</i>	<i>Byte hit ratio</i>
LRU	0.8%	0.6%
GDSF	35.6%	4.6%

Algoritme GDSF lebih baik dalam *hit ratio* yaitu sebanyak 35.6% berbanding 0.8% dan *byte hit ratio* sebesar 4.6% berbanding 0.6% dalam informasi data yang ditampilkan *squidclient*.

3.9.2 Hasil Pengujian Algoritme LRU dan GDSF (Squidanalyzer).

a. Algoritme LRU

Tabel 11. Hasil pengujian algoritme LRU (Squidanalyzer)

No.	Hasil Pengujian	Jumlah
1.	Jumlah permintaan http	24.864
2.	Jumlah <i>hit</i>	0.7% (187 <i>hit</i>)
3.	Jumlah <i>miss</i>	99.3% (24.677 <i>miss</i>)
4.	Jumlah Permintaan <i>Byte</i>	2.201.060.590
5.	Jumlah <i>byte hit</i>	0.3% (7.349.623)
6.	Jumlah <i>byte miss</i>	99.7% (2.193.710.967)

Dari tabel hasil pengujian algoritme *Least recently used* (LRU) dengan *squidanalyzer* diperoleh data permintaan http sebanyak 24.864 dengan jumlah *hit* sebesar 0.7% atau 187 *hit* dengan perbandingan jumlah *miss* 99.3% atau 24.677. jumlah permintaan *byte* sebesar 2.201.060.590 *byte*, *byte hit* sebesar 0.3% atau 7.349.623 *byte*, dan *byte miss* sebesar 99.7% atau 2.193.710.967 *byte*.



Gambar 10. Hasil pengujian algoritme LRU (Squidanalyzer)

b. Algoritme GDSF

Tabel 12. Hasil pengujian algoritme GDSF (Squidanalyzer)

No.	Hasil Pengujian	Jumlah
1.	Jumlah permintaan http	25.450
2.	Jumlah <i>hit</i>	23.9% (6088 <i>hit</i>)
3.	Jumlah <i>miss</i>	76.1% (19362 <i>miss</i>)
4.	Jumlah Permintaan <i>Byte</i>	2.237.675.843
5.	Jumlah <i>byte hit</i>	3.04% (68.029.816)
6.	Jumlah <i>byte miss</i>	96.96% (2.169.646.027)

Dari tabel hasil pengujian algoritme *Greedy dual size frequently* (GDSF) dengan *squidanalyzer* diperoleh data permintaan http sebanyak 25.450 dengan jumlah *hit* sebesar 23.9% atau 6088 *hit* dengan perbandingan jumlah *miss* 76.1% atau 19.362. jumlah permintaan *byte* sebesar 2.237.675.843 *byte*, *byte hit* sebesar 3.04% atau 68.029.816 *byte*, dan *byte miss* sebesar 96.96% atau 2.169.646.027 *byte*.



Gambar 11. Hasil pengujian algoritme GDSF (Squidanalyzer)

Tabel 13. Hasil Perbandingan Algoritme GDSF (Squidanalyzer)

Algoritme	Hit ratio	Byte hit ratio
LRU	0.7%	0.3%
GDSF	23.9%	3.04%

Algoritme GDSF lebih baik dari LRU dalam *hit ratio* yaitu sebanyak 23.9% berbanding 0.7% dan *byte hit ratio* sebesar 3.04% berbanding 0.3% dalam informasi data yang ditampilkan *squidanalyzer*.

4 Kesimpulan

Berdasarkan penelitian yang telah dilakukan oleh penulis, maka dapat ditarik kesimpulan sebagai berikut :

- 4.1 Proses membangun dan mengkonfigurasi proxy server dengan fungsi sebagai web caching dapat dilakukan dengan baik, serta berbanding lurus dengan teori yang menjadi rujukan dalam penelitian diantaranya mengenai hasil informasi yang diberikan *access.log*, dimana jika permintaan client tidak dapat dipenuhi oleh cache maka hasil outputnya TCP_MISS, UDP_MISS dll. dan jika permintaan dapat dipenuhi oleh cache maka hasil outputnya TCP_HIT, TCP_REFRESH_HIT dll.
- 4.2 Hasil dari implementasi algoritme least recently used (LRU) dan greedy dual size frequently (GDSF) menghasilkan data perbandingan yaitu LRU tidak sebaik GDSF karena angka hit dan byte hit yang dihasilkan algoritme LRU dibawah algoritme GDSF yaitu 187:6088, namun bukan berarti algoritme LRU tidak baik dalam cache replacement. Algoritme LRU akan berjalan lebih optimal jika diterapkan pada sebuah instansi yang pada aktivitasnya terdapat website favorit, contohnya jika pada sebuah universitas terdapat web favorit seperti e-perpustakaan, e-portofolio atau e-learning yang sering digunakan maka LRU akan dapat menjalankan tugasnya dengan baik karena konsep penghapusan LRU adalah berdasarkan recent atau data yang paling jarang digunakan.
- 4.3 Penelitian mengenai Analisis dan Implementasi Algoritme LRU dan GDSF Sebagai Aturan Cache Replacement Pada Proxy server dapat diterapkan dalam siklus hidup network development life cycle (NDLC) cisco PPDIIOO.

5 Referensi

- [1] APJII (Asosiasi Penyedia Jasa Internet Indonesia). 2017. Penetrasi dan Perilaku Pengguna Internet Indonesia. Survey 2017.
- [2] Nurwasito H. 2007. Analisa Algoritme Pergantian Cache Pada Proxy Web Server Internet dengan Simulasi. Jurnal EECCIS Vol. 1, No. 1.
- [3] DU Jianhai, GAO Shiwey, LV Jianghua, LI Qianqian, MA Shiong. 2018. A Web Cache Replacement Strategy for Safety-Critical System. Technical Gazette 25, 3, 820-830. ISSN 1330-3651(Print), ISSN 1848-6339 (Online).
- [4] Bruno A, Jordan S. 2011. Official Cert Guide CCDA 640-846. Learn , Prepare and Practice for Exam Success. USA. ISBN-10: 1-58714-257-0.
- [5] Susilo AD. 2013. Administrasi Server. Kementerian Pendidikan & Kebudayaan.
- [6] Rodeh O. 2012. BTRFS : The Linux B-tree Filesystem. IBM Research Report. USA. CA 95120-6099.
- [7] Wessels D. 2009. Squid The Definitive Guide. USA. ISBN-10 : 0-596-00162-2. ISBN-13 : 978-0-596-00162-9.
- [8] Sukaridhoto S, 2014. Buku Jaringan Komputer 1. Politeknik Elektronika Negeri Surabaya (PENS).
- [9] Towidjodjo R. 2014. Mikrotik Kung Fu kitab 1. Vol.4 No.1. ISSN: 1978-1261.
- [10] Ubuntu Team. 2014. Ubuntu Indonesia Linux Untuk Umat Manusia. Dalam <http://ubuntu-id.org>. diakses pada 16 September 2019.
- [11] Pangera Abas A, 2013. Algoritma Page Replacement. STMIK AMIKOM Yogyakarta.
- [12] Khasanah F N. 2017. Squid Proxy Server untuk Peningkatan Performa Akses Internet Pada Ubuntu Server 10.10. BINA INSANI ICT JOURNAL Vol. 4, No/ 1. ISSN: 2355-3421.
- [13] Yuisar, dkk.2015. Analisa Pemanfaatan Proxy Server Sebagai Media Filtering Dan Caching Pada Jaringan Komputer. Jurnal Media Infotama Vol. 11 No. 1. ISSN: 1858-2680.