

PENINGKATAN PERFORMANSI TRAFIK SMS A2P PADA SMSC DENGAN METODE PENAMBAHAN BUFFER DAN PERUBAHAN SKEMA PENGIRIMAN ULANG SMS

Nita Manutur S dan Irmayani

Email : ukksu.camp@gmail.com, ir.irmayani@istn.ac.id
Program Studi Teknik Elektro, Fakultas Teknologi Industri,
Institut Sains Dan Teknologi Nasional Jakarta**Abstrak**

Performansi trafik SMS A2P SOA SDC 83800 mengalami penurunan dikarenakan meningkatnya *error code 65 "Throttled for A#"* di sisi SMSC. Metode yang diterapkan untuk meningkatkan performansi tersebut ialah metode penambahan *buffer* dan perubahan skema pengiriman ulang SMS di sisi SMSC. Pemilihan metode ini dikarenakan *error code 65* yang berhubungan dengan *buffer* SMSC. *Error code 65* akan muncul jika jumlah *buffer* untuk SMS A2P SOA SDC 83800 sudah penuh.

Dari hasil pengecekan, ditemukan bahwa yang menjadi penyebab penuhnya *buffer* ialah meningkatnya trafik SMS yang dikirim oleh *user* SOA, dan meningkatnya *error absent subscriber*, *subscriber power off* dan *subscriber busy (user behavior)* di sisi FDA. Pemilihan metode ini di karenakan tidak memungkinkannya jika dilakukan pengurangan pengiriman trafik SMS dari sisi *user* SOA, dan tidak memungkinkannya juga untuk mengganti SMSC.

Adapun hasil yang telah dicapai dengan diterapkannya metode penambahan *buffer* dan perubahan skema pengiriman ulang sms di sisi SMSC, diantaranya adalah meningkatnya performansi trafik SMS A2P SOA SDC 83800 dengan hilangnya *error code 65 "throttled for A#"*, SMSC dapat menampung trafik SMS dari aplikasi/*user* SOA lebih banyak, dan dengan adanya perubahan skema pengiriman ulang SMS maka sms yang ditampung tidak akan disimpan terlalu lama di *buffer*.

Kata kunci : SMS, SMSC, *Buffer*, Kapasitas, Performansi, *Throttled***Abstract**

The performance of SOA A2P SMS traffic SDC 83 800 that declined due to increased error code 65 "throttled for A #" on the SMSC. The method is applied to improve the performance of the method is the addition of buffer and retransmission scheme SMS changes in the configuration of the SMSC. Selection of this method due to error code 65 relating to SMSC buffer. Where this error will appear if the number of buffers for A2P SMS SOA SDC 83 800 already full.

The results of checks, it was found that the cause of the full buffer is increased SMS traffic sent by the user SOA, and increased error absent subscriber, the subscriber power off and busy subscriber (user behavior) at the FDA. Selection of this method does not allow it in because if the reduction of SMS traffic delivery of SOA user side, and not allow it also to replace its SMSC.

The results that have been achieved with the application of the method of adding the buffer and change retransmission scheme sms in the configuration SMSC, include an increase in the performance of SMS traffic A2P SOA SDC 83 800 with the loss of error code 65 "throttled for A #", the SMSC can accommodate SMS traffic from the application / SOA user more, and with the change of retransmission scheme accommodated SMS, sms will not be stored too long in the buffer.

Keywords : SMS, SMSC, Buffer, Capacity, Performance, Throttled

I. Pendahuluan

Short Message Service (SMS) adalah salah satu tipe *Instant Messaging* (IM) yang memungkinkan pengguna untuk bertukar pesan singkat kapanpun, walaupun pengguna sedang melakukan panggilan atau akses data. Fitur SMS ini merupakan fitur yang sering digunakan selain penggunaan panggilan telepon.

Namun, permasalahan yang muncul adalah ketika pesan yang dikirim tidak diterima.

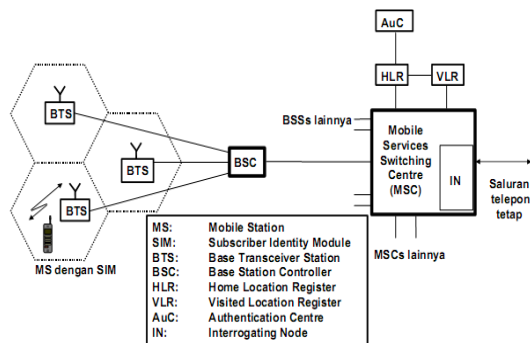
Dalam makalah ini akan dibahas mengenai penanganan terhadap turunnya performansi trafik SMS A2P dari aplikasi SOA dengan SDC 83800 dikarenakan naiknya *error code 65 "Throttled for A#"*, dengan metode penambahan *buffer* dan perubahan skema pengiriman ulang sms untuk SMS A2P SDC 83800 di

sisi konfigurasi SMSC, tanpa mengurangi jumlah trafik yang dikirim dari sisi aplikasi dan tetap menggunakan SMSC yang sama.

II. Landasan Teori

2.1 Arsitektur Jaringan GSM

Secara umum, *network element* dalam arsitektur jaringan GSM dapat dibagi menjadi 3 elemen yaitu *Base Station Sub-sistem* (BSS), *Network Sub-System* (NSS), *Operation and Support Sistem* (OSS) [4].



Gambar 2.1 Arsitektur GSM

2.1.1 Base Station Sub-system (BSS)

Secara umum, *Base Station Sub-sistem* terdiri dari BTS (*Base Transceiver Station*) dan BSC (*Base Station Controller*).

BTS adalah perangkat GSM yang berhubungan langsung dengan MS. BTS berhubungan dengan MS melalui *air interface* atau disebut juga *Um Inteface*. BTS berfungsi sebagai pengirim dan penerima (*transceiver*) sinyal komunikasi dari/ke MS yang menyediakan *radio interface* antara MS dan jaringan GSM. Karena fungsinya sebagai *transceiver*, maka bentuk fisik sebuah BTS adalah tower dengan dilengkapi antena sebagai *transceiver*.

2.1.2 Network Sub-System (NSS)

Network Sub-System terdiri dari MSC, HLR, VLR, AuC dan EIR.

MSC adalah *network element central* dalam sebuah jaringan GSM. Semua hubungan (*voice call/transfer data*) yang dilakukan oleh mobile subscriber selalu menggunakan MSC sebagai pusat pembangunan hubungannya.

HLR adalah *network element* yang berfungsi sebagai sebuah *database* untuk menyimpan semua data dan informasi mengenai pelanggan yang tersimpan secara permanen, dalam arti tidak tergantung pada posisi pelanggan. HLR bertindak sebagai pusat informasi pelanggan yang setiap waktu akan diperlukan

oleh VLR untuk merealisasi terjadinya komunikasi pembicaraan.

VLR adalah *network element* yang berfungsi sebagai sebuah *database* yang menyimpan data dan informasi pelanggan, dimulai pada saat pelanggan memasuki suatu area yang bernaung dalam wilayah MSC. VLR bertindak sebagai *database* pelanggan yang bersifat dinamis, karena selalu berubah setiap waktu, menyesuaikan dengan pelanggan yang memasuki atau berpindah dalam suatu area cakupan suatu MSC.

AuC menyimpan semua informasi yang diperlukan untuk memeriksa keabsahan pelanggan, sehingga usaha untuk mencoba mengadakan hubungan pembicaraan bagi pelanggan yang tidak sah dapat dihindarkan.

EIR memuat data-data peralatan pelanggan (*Mobile Equipment*) yang diidentifikasi dengan IMEI (*International Mobile equipment Identity*).

2.1.3 Operation and Support System (OSS)

Operation and Support System (OSS) sering juga disebut dengan OMC (*Operation and Maintenance Center*) adalah sub sistem jaringan GSM yang berfungsi sebagai pusat pengendalian dan *maintenance* perangkat (*network element*) GSM yang terhubung dengannya.

2.2 Short Message Service (SMS)

SMS merupakan salah satu fitur teknologi seluler berupa layanan pengiriman pesan teks pendek antar telepon seluler. SMS menjadi salah satu media komunikasi antar individu yang berkembang dengan pesat. Definisi SMS sebagai mekanisme pengiriman pesan pendek melalui jaringan mobile [3].

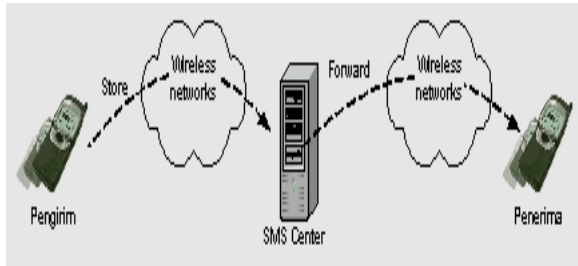
Layanan SMS merupakan sebuah layanan yang bersifat *non real time* dimana sebuah *short message* dapat di-*submit* ke suatu tujuan yang sedang aktif atau tidak. Bila dideteksi bahwa tujuan tidak aktif, maka sistem akan menunda pengiriman ke tujuan hingga tujuan aktif kembali, kecuali bila diberlakukan aturan bahwa *short message* yang telah melampaui batas waktu tertentu harus dihapus dan dinyatakan gagal terkirim.

2.3 SMSC

SMS Center (SMSC) merupakan pengelola dan pentransmisi SMS. SMSC berupa *hardware* dan *software* yang berfungsi sebagai kantor pos dalam mengatur lalu lintas surat [11]. Setiap jaringan GSM memiliki satu atau lebih SMSC untuk menyortir dan melakukan perutean (*routing*) pesan. SMSC bertugas melakukan pengecekan, pengaturan, dan pengiriman pesan kepada operator. Selain itu SMSC juga dapat mengirim dan menerima pesan antar jaringan GSM.

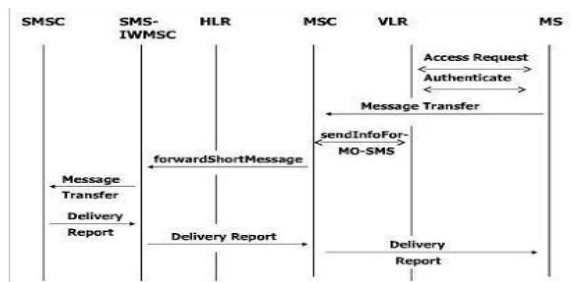
Mekanisme pengiriman pesan SMS secara store and forward berarti pengirim pesan SMS menuliskan pesan dan nomor telepon tujuan, kemudian

mengirimkannya (store) ke server SMSC yang kemudian bertanggung jawab untuk mengirimkan pesan tersebut (forward) ke nomor telepon tujuan, seperti terlihat pada gambar 2.2.



Gambar 2.2 Mekanisme pengiriman pesan SMS [2]

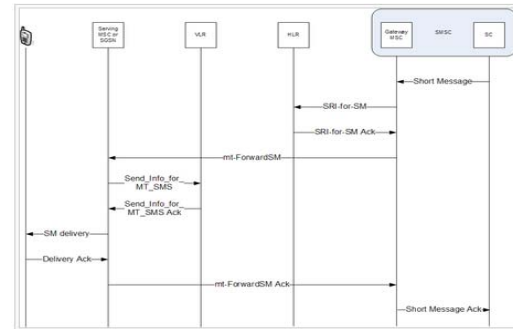
Berdasarkan pengirimnya, dikenal dua jenis SMS, yakni SMS P2P (Person to Person) dan SMS A2P atau P2A (Application to Person atau Person to Application). Prosedur dasar SMS P2P dalam SMSC adalah MO, dan MT. Proses MO adalah pengiriman pesan dari MS ke SMSC dan mengirimkan kembali laporan ke MS tentang hasil pengiriman pesan tersebut.



Gambar 2.3 Proses Pengiriman SMS MO [1]

Proses pengiriman MO pada gambar 2.3 menjelaskan bahwa setelah terjadi proses request dan autentikasi antara MS dan VLR, pesan yang dikirim MS terkirim ke MSC. MSC akan melakukan verifikasi terhadap pesan yang dikirim ke VLR (sendInfoForMO-SMS). Kemudian MSC mengirim pesan ke SMS-IWMSC (Inter-Working MSC for SMS) menggunakan operasi "Forward Short Message". Lalu SMS-IWMSC mengirimkan message ke SMSC (Short Message Service Centre). SMSC melakukan ack bahwa pesan sudah diterima oleh SMSC ke MSC. Dan terakhir, MSC mengirimkan status pengiriman pesan (Delivery Report) ke MS.

Proses MT merupakan pengiriman SMS dari SMSC ke MS. Untuk pengiriman SMS ini akan disediakan informasi pengiriman, baik delivery report untuk SMS yang berhasil maupun failure report untuk pengiriman yang gagal karena sebab tertentu, sehingga memungkinkan SMSC untuk melakukan pengiriman ulang.



Gambar 2.4 Proses pengiriman SMS MT [12]

Gambar 2.4 menjelaskan proses pengiriman SMS MT. Setelah SMSC menerima pesan yang dikirim MS A, kemudian SMSC mengirimkan pesan ke SMS-GMSC. SMS-GMSC akan menginterogasi HLR untuk informasi routing, HLR membalas informasi routing ke SMS-GMSC. SMS-GMSC meneruskan pesan ke MSC/VLR. MS di-paging dan koneksi terbentuk antara MS dan network, sebagaimana dalam setup panggilan normal. (Dengan demikian posisi MS diketahui dan apakah MS boleh berada dalam network / proses otentikasi). Jika otentikasi berhasil, MSC/VLR mengirim pesan sms tersebut ke MS. SMS dikirim melalui kanal signaling SDCCH). Jika pengiriman berhasil, delivery report dikirim dari MSC/VLR ke SMSC. Namun jika tidak, MSC/VLR akan menginformasikan ke HLR, dan failure report dikirim ke SMS-C. Pada kasus pengiriman yang gagal, HLR dan VLR akan mendapat informasi "Messages waiting" yang menunjukkan ada pesan di SMSC yang menunggu untuk dikirimkan ke MS. Informasi di HLR terdiri dari list SMSC pengirim pesan, sedangkan di VLR terdapat "flag" yang menunjukkan apakah list pesan dalam keadaan kosong atau tidak. Jika MS available dan siap menerima pesan, maka HLR akan memberitahu SMSC.

Untuk proses pengiriman SMS P2A atau A2P sama saja dengan proses MO dan MT pada SMS P2P, dibedakan dengan adanya aplikasi setelah SMSC.

2.4 Alur Pengiriman SMS

Elemen-elemen utama pada arsitektur SMS terdiri dari SMS MO, SMSC, FDA, SMS MT, SDC. Short Message Service Center (SMSC) yang berfungsi mengelola dan mentransmisikan SMS. Di SMSC juga terkonfigurasi retry scheme, yaitu skema pengiriman ulang pesan yang gagal dikirim saat pengiriman pertama sehingga pesan tersebut dikirimkan beberapa kali sampai berhasil atau sampai limit maksimal skema

pengiriman. Selain *retry scheme*, di SMSC juga terkonfigurasi *buffer*, yaitu tempat penyimpanan data sementara. *Buffer* di SMSC terletak di bagian memori. Besarnya *buffer* ditentukan berdasarkan SDC (jika tidak ditentukan, maka besarnya *buffer* akan menggunakan *default*) dan disesuaikan dengan kapasitas dari memori atau *buffer server SMSC*. Total *buffer* di SMSC saat ini adalah satu juta sms.

Untuk sistem antrian yang digunakan oleh SMSC adalah FIFO (*First In First Out*). Teknik antrian FIFO mengacu pada FCFS (*First Come First Served*), yakni paket data atau sms yang pertama datang, diproses terlebih dahulu.

Pada SMSC, sms kedua akan dikirim setelah sms pertama berstatus sukses (terkirim ke penerima) atau *expired* (berdasarkan *validity period time*).

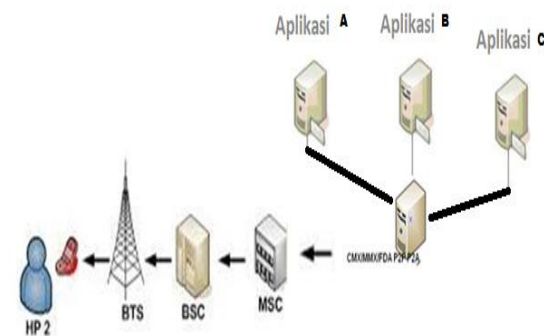
Pada suatu operator, terdapat sebuah server yang disebut FDA (*First Delivery Attempt*), yaitu perangkat yang pertama kali menerima sms dan yang pertama kali mencoba mengirim sms ke pelanggan atau aplikasi yang di tuju. FDA ini tidak dilengkapi dengan penyimpanan SMS sehingga SMS yang gagal pada pengiriman yang pertama akan dikirim ke SMSC untuk dikirimkan kembali melalui skema pengiriman ulang. Selain sebagai pengirim pertama, FDA juga sebagai tempat *charging* (karena FDA terkoneksi ke *billing*). Alasan operator menggunakan server ini adalah sebagai *load balancing*. Supaya SMSC tidak bekerja sendirian sebagai server pengirim SMS. Sehingga tugas SMSC hanya sebagai pengirim ulang SMS yang gagal dikirim oleh FDA.

Short Destination Code merupakan identitas atau nomor pengirim dari suatu aplikasi ketika mengirim SMS ke pelanggan. Misalnya untuk mengirim SMS tentang promo RBT (*ring back tone*), menggunakan SDC 1818.

III. Deskripsi Pembahasan

3.1 Alur Pengiriman SMS A2P

Untuk pengiriman SMS A2P (*application to person*) terdapat dua alur, yakni pengiriman tanpa

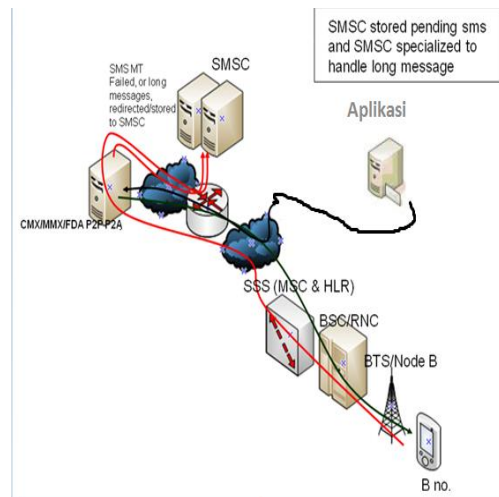


SMSC dan dengan SMSC.

Gambar 3.1 Alur Pengiriman SMS A2P tanpa SMSC

Pada gambar 3.1 terlihat proses pengiriman SMS dari beberapa aplikasi menuju nomor HP2. Ketika mengirim pesan (SMS MO), maka aplikasi akan mengirim pesan tersebut ke FDA, lalu dari FDA akan dikirim ke MSC, BSC, BTS, dan terakhir ke nomor HP2. Pada pengiriman pertama, HP2 mengirimkan sinyal yang kemudian diterjemahkan oleh FDA untuk mengetahui *error code* untuk status nomor HP2. Untuk gambar 2.5 terlihat pesan terkirim pada pengiriman pertama, artinya sinyal yang diterima oleh FDA diterjemahkan bahwa nomor HP2 *available* dan bisa untuk menerima pesan. Jadi pesan tersebut berhasil terkirim pada pengiriman pertama tanpa dikirim ke SMSC.

Sedangkan pada gambar 3.2, merupakan proses pengiriman SMS jika nomor B mengirimkan sinyal yang diterjemahkan oleh FDA sebagai *error code* (*absent subscriber, subscriber busy, dll*) bahwa nomor B tidak dapat menerima SMS, maka SMS dikirim ke SMSC (SMS *pending*) untuk selanjutnya dikirim kembali dengan skema pengiriman ulang SMS yang telah dikonfigurasi di SMSC.

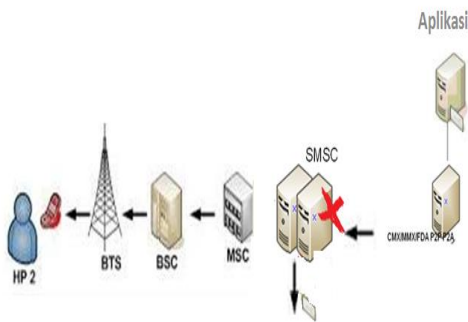


Gambar 3.2 Alur Pengiriman SMS A2P dengan SMSC

SMS *pending* dikirimkan kembali oleh SMSC dengan skema *retry* yang sudah dikonfigurasi di SMSC. SMS yang dikirimkan oleh SMSC ke nomor B akan terus dikirim ulang oleh SMSC sampai diterima sinyal dari nomor B yang diterjemahkan oleh SMSC bahwa

nomor B telah *available* untuk menerima SMS dari SMSC.

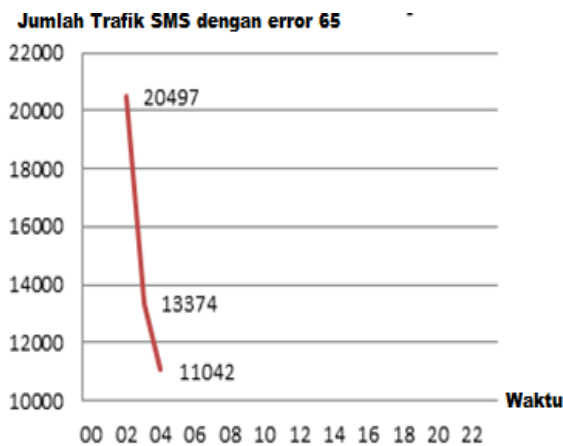
Hal yang dibahas dalam makalah ini adalah *error* “*throttled for A#*” dengan *error code* 65 yang ditemukan di SMSC saat aplikasi SOA mengirim SMS. *Error 65 Throttled for A#* merupakan *error* yang muncul ketika jumlah SMS yang dikirim melebihi kuota/*throttled*. Munculnya *error* ini disebabkan oleh *buffer* di SMSC untuk aplikasi tersebut sudah penuh. Jadi *error* tersebut adalah *error* dari SMSC, bukan dari penerima. Dengan munculnya *error* ini, maka SMS tidak akan diproses, melainkan langsung dibuang (Gambar 3.3).



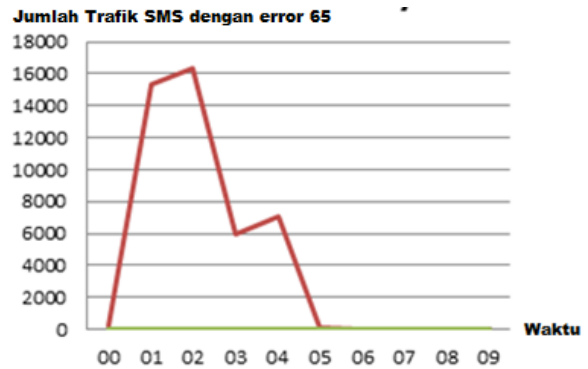
Gambar 3.3 Alur Pengiriman SMS A2P dengan *error*

3.2 Data Performansi Trafik SMS A2P SDC 83800

Dari alur pengiriman SMS A2P yang sudah dijelaskan sebelumnya, saat FDA mengirim trafik ke SMSC, SMSC akan mengirim *acknowledge* (ack) ke FDA terhadap trafik yang dikirim. Kemudian FDA akan mengirim ack tersebut ke aplikasi. Aplikasi akan melihat apakah trafik yang dikirim mendapatkan respon atau tidak. Dari data yang telah dikumpulkan oleh aplikasi, terlihat banyak trafik sms yang mendapat ack *negative*.

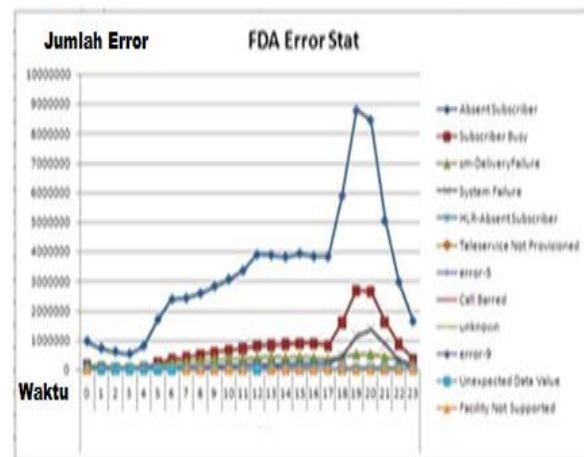


Gambar 3.4 Grafik data trafik sms dgn *error code* 65 hari ke-19



Gambar 3.5 Grafik data trafik sms dengan *error code* 65 hari ke-20

Gambar 3.4 dan 3.5 merupakan *sample* grafik persebaran *error code* 65 di sisi aplikasi. Jumlah trafik sms dengan *error* pada hari ke-19 adalah 44.913 dan pada hari ke-20 adalah 44.630. Setelah aplikasi mengetahui bahwa trafik yang dikirim mendapat ack *negative*, maka dilakukan pengecekan dari sisi FDA.



Gambar 3.6 Grafik statistik *error* FDA

Dari data *sample error statistic* FDA (gambar 3.6), dapat dilihat bahwa *error* yang paling banyak muncul terjadi pada hari ke-19. *Error* tersebut ialah *absent subscriber*, *subscriber power off* dan *subscriber busy*. Ketiga *error* tersebut merupakan penyumbang terbesar kegagalan SMS yang dikirim oleh FDA pada pengiriman pertama. Sesuai flow pengiriman SMS A2P pada suatu operator, maka sms yang gagal akan dikirim ke SMSC untuk dikirim kembali. Kemudian dilakukan pengecekan dari sisi SMSC.

Penarikan data dari sisi SMSC dilakukan dengan mengetikkan *command* (gambar 3.7) di server SMSC.

```
<382CBTSMSC1:/home/smc/billdata/bpsbill/11/backu
p>CdrReaderMO.sh

put CDR below, then hit <<enter>>:

410923553,83800,6283185103891,,62818445009,0,0,0,
0,0,0,0,0,,0,153,0,65,0,2018/05/18
12:43:05,,0,soa,,,,,,,,,,,,,Gateway_G,,,,,,,,,,,,
'''

Originator Addr      : 83800
Destination Addr     : 6283185103891
Result                : 65
Original MO Submit Time : 2018/05/18
12:43:05
```

Gambar 3.7 *command* dan *log* hasil penarikan data pada server

Dari *sample log* SMSC gambar 3.7, ditemukan SMS A2P dari SDC 83800 ke 6283185103891 pada hari ke-18 mendapat *result* 65. *Result* disini menandakan bahwa SMS A2P tersebut mendapat *error code* 65 yakni *Maximum Submission Number Exceeded* atau biasa disebut “*Throttled for A#*”. Kemudian keseluruhan *log* SMS A2P dari SDC 83800 di server SMSC di kumpulkan untuk melihat *error code* apa saja yang ditemukan pada SMS A2P dari SDC 83800. Rumusan perhitungan jumlah *error code* 65 di SMSC, di dapatkan dari hasil keluaran *command* (gambar 3.8).

```
<406
CBTSMSC1:/home/smc/billdata/bpsbill/11/buffer>cat
prm* | awk -F " , " '{if ($2 == 83800) print $0}'
```

Gambar 3.8 *command* hasil total *error*

Pada gambar 3.9 merupakan *command* sample untuk mendapatkan trafik *error* sebelum trafik *error* di awk-F (dijumlahkan)

```
<408
CBTSMSC1:/home/smc/billdata/bpsbill/11/buffer> cat
prm* , " '{if ($2 == 83800) print $0}' | more

1179010532,83800,6283107733353,,62818445009,0,0,
0,0,0,1,0,0,0,,0,153,0,65,0,2018/05/15
09:58:17,,0,soa,,,,,,,,,,,,,Gateway_G,,,,,,,,,,,,
'''

1179010533,83800,6283808563214,,62818445009,0,0,
0,0,0,1,0,0,0,,0,153,0,65,0,2018/05/15
09:59:10,,0,soa,,,,,,,,,,,,,Gateway_G,,,,,,,,,,,,
'''

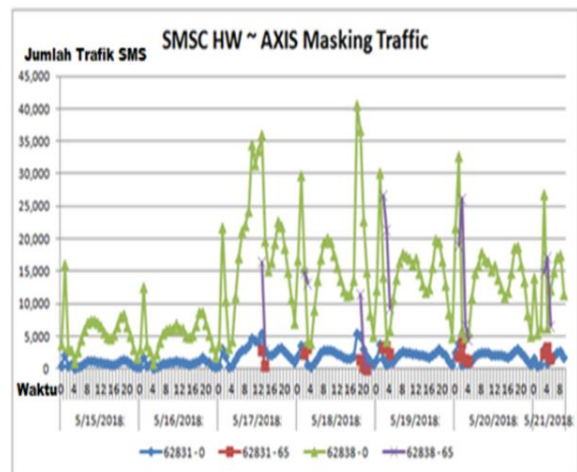
1179010534,83800,6283186590023,,62818445009,0,0,
0,0,0,1,0,0,0,,0,153,0,65,0,2018/05/15
10:05:17,,0,soa,,,,,,,,,,,,,Gateway_G,,,,,,,,,,,,
'''

1179010535,83800,6283155339001,,62818445009,0,0,
0,0,0,1,0,0,0,,0,153,0,65,0,2018/05/15
10:21:14,,0,soa,,,,,,,,,,,,,Gateway_G,,,,,,,,,,,,
'''

<408
CBTSMSC1:/home/smc/billdata/bpsbill/11/buffer>
```

Gambar 3.9 *Command* trafik *error* sebelum awk-F

Grafik pada gambar 3.10 merupakan hasil penarikan data dari SMSC setelah dilakukan penjumlahan (awk -F) yang dirangkum yaitu grafik Trafik SMS untuk nomor dengan prefix ‘6283’ dari SDC 83800.



Gambar 3.10 Grafik Trafik SMS nomor prefix ‘6283’ dari SDC 83800

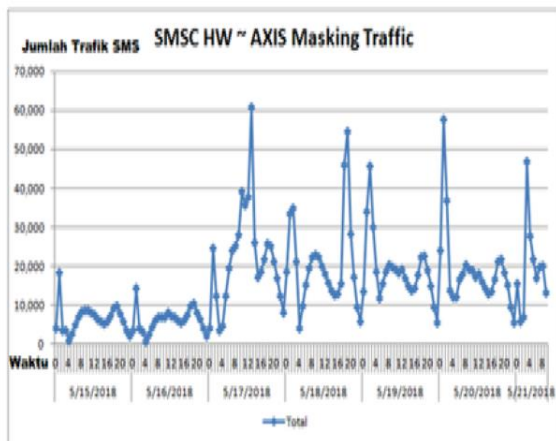
Dari Grafik di gambar 3.10 diperoleh informasi pada hari ke-17 terjadi peningkatan *attempt* ke arah SMSC untuk nomor dengan prefix ‘6283’.

Seiring dengan peningkatan tersebut maka terjadi pula peningkatan *error code 65 (throttling for A#)* pada jam-jam tertentu, sebagian besar saat dimana *traffic attempt*-nya tinggi. *Error code 65* ini berhubungan dengan *buffer* SMSC. Ketika *buffer* masih menahan trafik hijau, dan belum selesai proses pengirimannya, trafik ungu yang hendak masuk tidak bisa ditampung lagi. Sehingga trafik ungu tersebut mendapat *error code 65*. Detail penjelasan tentang *error code 65* dapat dilihat pada lampiran.

Ditemukan bahwa *buffer* SMSC untuk SMS A2P dari SDC 83800 penuh, sehingga SMS A2P SDC 83800 yang masuk ke SMSC tidak dikirimkan ke pelanggan.

Solusi yang diambil untuk mengurangi *error code 65* ini ialah dengan menaikkan jumlah maksimum *throttling* SMS A2P SDC 83800 di SMSC dan mengubah skema pengiriman ulang sms. Alasan dipilihnya metode ini ialah karna tidak memungkinkannya dilakukan metode pengurangan jumlah pengiriman trafik dari sisi aplikasi, mengingat akan ber-*impact* ke *revenue* operator. Dan untuk SMSC yang digunakan tidak dapat diganti (tetap menggunakan SMSC yang sama).

Metode menaikkan *buffer* dan perubahan skema pengiriman ulang untuk SMS A2P SDC 83800 akan dilakukan di sisi konfigurasi SMSC. Besarnya jumlah *buffer* yang akan dinaikkan bisa ditentukan berdasarkan *counting hourly volume* SMS dari SDC 83800 yang masuk ke SMSC.

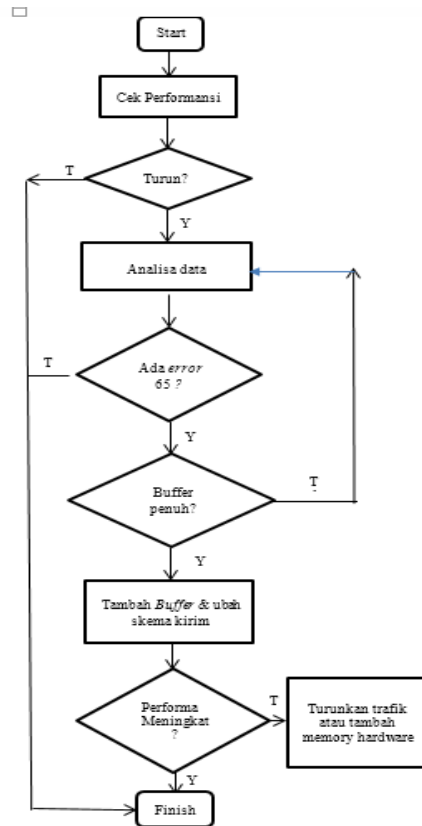


Gambar 3.11 Grafik total *Hourly traffic* sms SMSC ke nomor prefix ‘6283’

Gambar 3.11 merupakan data total *hourly traffic* untuk SMS yang masuk ke SMSC dari SDC 83800 ke nomor dengan prefix ‘6283’. Untuk *peak hour* bisa mencapai 60k *attempt/hour*, kenaikannya mencapai 2-3 kali lipat dari *traffic* sebelumnya.

3.3 Skema Pengiriman Ulang SMS

Untuk skema pengiriman ulang sms di SMSC ada dua skema yakni *default scheme* dan 818 *retry scheme*. Kedua skema ini di set berdasarkan *error* yang terjadi. Perbedaannya ialah interval waktu pengiriman sms. Ketika masalah performansi ini terjadi, skema pengiriman ulang SMS A2P SDC 83800 ialah menggunakan *default scheme*. Dan untuk menaikkan performansinya, skema pengiriman ulang SMS A2P SDC 83800 akan diubah menjadi 818 *retry scheme*.



Gambar 3.12 Diagram Alir Implementasi

Gambar 3.12 merupakan diagram alir implementasi yang menjelaskan prosedur yang dilakukan untuk menaikkan performansi, seperti ditemukan turunnya performansi SMS A2P SOA SDC 83800 (berupa grafik kenaikan *error*) dari sisi aplikasi, kemudian mendata *error* yang menyebabkan turunnya performansi SMS A2P SOA SDC 83800 (pengecekan dilakukan disisi FDA dan SMSC), dari hasil pengecekan ditemukan *error code 65 “Throttled for*

A#” disisi SMSC adalah sebagai penyumbang tertinggi yang menyebabkan turunnya *success rate* pengiriman SMS A2P SOA SDC 83800. Prosedur selanjutnya ialah menganalisa penyebab munculnya *error* tersebut, dan ditemukan bahwa *buffer* SMSC penuh, sehingga SMS A2P SOA SDC 83800 yang masuk ke SMSC tidak bisa dikirimkan ke pelanggan, dan metode yang diambil ialah penambahan *buffer* dan merubah skema pengiriman ulang SMS A2P SOA SDC 83800 di sisi konfigurasi SMSC. Sebelum metode ini diterapkan, harus dicek terlebih dahulu, apakah jumlah *buffer* untuk SMS A2P SOA SDC 83800 masih mencukupi untuk diperbesar jumlahnya. Jika masih mencukupi, maka metode ini bisa diterapkan. Dan dari hasil pengecekan, jumlah *buffer* nya masih cukup, maka metode ini bisa diterapkan. Setelah metode ini diterapkan, dilakukan pengecekan kembali di sisi SMSC apakah performansi *success rate* SMS A2P SDC 83800 sudah naik, dan apakah masih ditemukan *error* “*Throttled for A#*”. Setelah dilakukan penambahan *buffer* dan merubah skema pengiriman ulang di SMSC, terlihat performansi SMS A2P SDC 83800 kembali normal tanpa adanya *error* ”*Throttled for A#*”.

3.4 Penambahan Buffer dan perubahan skema pengiriman ulang SMS

Berdasarkan data *hourly traffic* SMS A2P SDC 83800, dimana terlihat besarnya kenaikan trafik adalah 60.000 *attempt/hour* (kenaikannya mencapai 2-3 kali lipat dari trafik sebelumnya) sedangkan jumlah *buffer* yang bisa menampung adalah 10.000 *attempt* (sesuai dengan *default*),

[1] SMSC Basic Config Management					
Section Name	Key Name	Key Description	Value	Effective in Packet	Remark
Common	Mail/SMS/Retry-Service	Default max submission count	10000	Yes	Default maximum submission count of the service entities.

Gambar 3.13 Jumlah Awal *Buffer* untuk SMS A2P SDC 83800

Maka jumlah *buffer* yang akan dinaikkan di sisi konfigurasi SMSC ialah sesuai dengan jumlah maksimum *buffer* yang telah disediakan untuk tiap SDC, yakni sebesar 150.000 *attempt* (hal ini untuk menghindari jika suatu saat jumlah trafik yang dikirim menjadi lebih besar lagi),

Service Number Head /	Sub-SME Number Head	Match Length	Max. MO SM Number	Max. MT SM Number
83800		5	150000	150000
83801		5	150000	150000
83802		5	150000	150000

Gambar 3.14 Jumlah *buffer* untuk SMS A2P SDC setelah diubah

Setelah dilakukan penambahan jumlah *buffer*, skema pengiriman ulang juga diubah, menjadi sbb untuk mengurangi jumlah antrian di SMSC (agar sms tidak terlalu lama di simpan oleh SMSC), maka skema pengiriman ulang SMS A2P SDC 83800 juga diubah menjadi 818 *retry scheme*, dimana sebelumnya menggunakan *default retry scheme* (gambar 3.15).

Attribute name of SM	Attribute value of SM
SM ID	450692777
Original Address	A005
Destination Address	6283162006841
MDMSC Address	62818445917
MTMSC Address	628184420312
Submit time	2018/05/21 22:00:03
The Final Time SM Delivery(MEMORY)/The T	2018/05/21 22:00:36
SM Status	1:Delivering succeeded
Error Code	0:Default
Delivery Attempt Times	2
Is StatusReport	No(0)
Called IMSI	510080354769394
Previous Error Code	0
Schedule Strategy Name	818 Retry Schema
Submitting Account	Gateway_G
Delivering Account	Gateway_G

Gambar 3.15 Tampilan perubahan skema 818

IV. Pengujian dan Analisis

Ada beberapa hal yang menjadi penyebab penuhnya *buffer* di SMSC dalam penelitian ini, antara lain yakni meningkatnya jumlah trafik. Jumlah trafik yang dikirim aplikasi SOA dengan SDC 83800 meningkat seiring dengan naiknya jumlah pelanggan pada operator tersebut. Kenaikan trafik menjadi 2 sampai 3 kali lipat dari biasanya. Sedangkan jumlah *buffer* di SMSC untuk menampung trafik sms dari aplikasi tersebut ialah 10.000 sms. Ketika *buffer* SMSC penuh, salah satu solusi yang diambil ialah menurunkan jumlah pengiriman trafik. Namun hal ini tidak dilakukan, mengingat akan adanya *impact* ke *revenue* operator. Ketika *buffer* dinaikkan maka trafik 8 ias diterima dan diproses kembali. Berikut detail metode penambahan *buffer* dan perubahan skema pengiriman ulang sms dalam kasus ini,

- Dari performansi trafik SMS A2P SDC 83800 terlihat adanya kenaikan *error* 65 “*throttled for A#*”. *Error* ini merupakan *error* yang mengindikasikan bahwa *buffer* atau kapasitas dari *virtual memory* yang dimiliki oleh SMSC untuk SDC 83800 sudah penuh. Sehingga sms selanjutnya tidak dapat ditampung lagi sehingga dibuang.

- Dalam metode ini, *buffer* tidak melakukan proses apa-apa untuk menghilangkan *error* 65. Karna *buffer* merupakan tempat penyimpanan sms sementara (*virtual memory*).

- *Retry schedule* diperbanyak dan dipercepat intervalnya (perubahan skema pengiriman ulang sms), agar *Success Rate* diharapkan lebih bagus dan agar *message* tidak terlalu lama disimpan di *L1 memory (physical memory)*.

- Menambah *max buffer*, agar SMSC 9 ias menampung lebih banyak antrian/*queue message* dari A# yang sama. Dalam hal ini A# adalah SDC 83800. Jadi *buffer* itu 9ias juga disebut dengan *queue/antrian*.

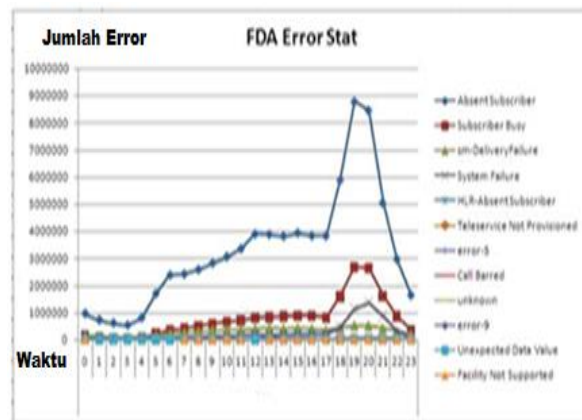
- Setelah dilakukan penambahan *buffer* dan perubahan skema pengiriman ulang SMS A2P SDC 83800 di sisi konfigurasi SMSC, maka terlihat performansi trafiknya menjadi normal kembali (ditandai dengan hilangnya *error 65*).

Penyebab kedua dari penuhnya *buffer* di SMSC dalam penelitian ini ialah meningkatnya *error absent subscriber, subscriber power off* dan *subscriber busy* di sisi FDA. *Error code absent subscriber, subscriber power off* dan *subscriber busy* merupakan *error code* yang berarti penerima dalam keadaan sibuk, *handset* mati, berada diluar jaringan, atau yang lainnya yang menyebabkan sms tidak 9ias dikirim dan tidak akan dibahas lebih lanjut. Dari sample log di FDA pada gambar 4.1,

5/20/18	12:43:03.000	83800	6283185103891	Paket HotRod3G+ Harian Rp1rb, 5MB Anda terhenti krn kartu Anda berada dlm masa tenggang. Silahkan isi pulsa & beli pakatnya di *123*400#. Info 817	84SMSC	ASP SOA
---------	--------------	-------	---------------	--	--------	------------

Gambar 4.1 Sample log sms gagal terkirim di FDA

Error code 84 merupakan *error code delivery failure* sehingga sms tersebut dikirim ke SMSC untuk dikirim ulang. Penyebab gagal tersebut (*delivery failure*) dirangkum pada grafik gambar 4.1.



Grafik 4.2 Error statistic FDA

Ketiga *error* tersebut merupakan penyumbang terbesar kegagalan pengiriman sms di sisi FDA dalam penelitian ini. *Error* ini termasuk dalam kategori *user behavior error* (dimana kegagalan pengiriman disebabkan oleh *error* di sisi pelanggan) bukan *system error*. Sesuai dengan alur pengiriman SMS A2P di operator tersebut, dimana jika FDA gagal mengirim sms ke penerima maka FDA akan mengirim sms tersebut ke SMSC. Dan SMSC akan mengirim ulang sesuai dengan skema pengiriman ulang yang telah terkonfigurasi. Namun jika jumlah sms yang dikirim oleh FDA sangat banyak, hingga melebihi batas *buffer* di SMSC, maka antriannya akan penuh dan tidak menutup kemungkinan trafik selanjutnya tidak akan diterima lagi oleh SMSC (dibuang).

Detail antrian dan pengiriman sms di SMSC terdiri dari proses MO (proses dimana SMSC menerima sms dari FDA), ketika SMSC menerima sms dari FDA, SMSC akan melakukan beberapa pengecekan (*authentication, blacklist rules, anti spam rules, etc*). Jika sudah ok, maka SMSC akan mengirim ACK ke FDA.

Setelah itu proses MT (proses dimana SMSC mengirim SMS yang telah diterima dari FDA ke pelanggan), pertama-tama SMSC akan menyimpan sms tersebut di *L1 memory (physical memory)*, kemudian SMSC akan mencoba mengirim sms ke pelanggan. Sebelum mengirim *real message*, SMSC akan mengirim SRI (*Send Routing Information*) ke HLR (*Home Location Register*) untuk mengetahui kondisi penerima (posisi, handset aktif atau tidak, dan lainnya). Jika HLR memberi *response* OK, maka SMSC akan mengirim *real message* ke penerima dengan kondisi:

- Jika sms berhasil dikirim ke penerima, maka *message* di *drop* dan dicatat sebagai *success transaction* di Database dan CDR (*Call Data Record*) SMSC.
- Jika sms tidak berhasil dikirim ke penerima, maka *message* tetap di simpan di *L1 memory (physical*

memory) dan menunggu jadwal pengiriman sms seperti yang ada di skema pengiriman ulang.

- Jika sampai beberapa kali pengiriman ulang tetap tidak berhasil terkirim atau sudah melebihi *expired time*, maka sms akan di drop dari L1 memory dan ditulis sebagai *failed transaction* di Database dan CDR SMSC.

- Jika utilisasi/kapasitas L1 memory sudah penuh, maka sms selanjutnya akan disimpan di L2 memory (logical memory yang di create di Hard disk).

- Message yang di simpan di L2 memory tidak akan di proses oleh SMSC sebelum ada *space* kosong di L1 memory.

- Dalam kasus ini, trafik SMS A2P SDC 83800 yang dikirim ke SMSC tidak dapat ditampung lagi, walaupun sudah disimpan di L2 memory. Sehingga sms yang baru datang akan di drop oleh L1 memory.

- Jadi ketika L1 memory untuk A# (SDC 83800) tersebut penuh, sms yang baru datang akan di *reject* oleh SMSC.

- Kemudian dilakukan penambahan buffer/memory di sisi konfigurasi SMSC agar trafik SMS A2P SDC 83800 dapat ditampung kembali.

- Dan juga dilakukan perubahan skema pengiriman ulang untuk SMS A2P SDC 83800 agar L1 memory tidak terlalu lama menyimpan smsnya (interval waktu pengiriman sms nya dipercepat).

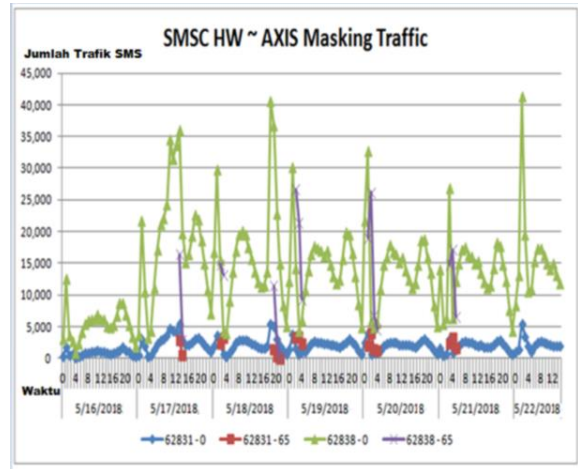
Untuk sistem antrian, SMSC menggunakan sistem FIFO (First In, First Out). Dimana ada dua jenis antrian yaitu :

- Antrian Normal (dimana penerima bisa menerima sms). Antrian ini terjadi ketika ada sms dari satu atau lebih pengirim (A#) ke penerima (B#) yang sama, dan pada waktu itu kondisi penerima bisa menerima sms tersebut.

- Antrian dimana penerima tidak bisa sms. Antrian ini terjadi ketika ada sms dari satu atau lebih pengirim ke penerima yang sama, dan pada waktu itu kondisi penerima sedang tidak bisa menerima sms tersebut (mungkin *power off*, atau diluar *service area*), maka sms tersebut akan di simpan di *memory* SMSC (*queue*) dan SMSC akan mengirim ulang sms pertama saja, sedangkan sms berikutnya akan terkirim jika sms pertama berubah statusnya (*success/failed/expired*).

Dalam penelitian ini metode mengganti SMSC tidak dapat dilakukan, mengingat operator tidak memiliki server SMSC dalam jumlah yang banyak untuk SMS A2P. Sehingga metode yang bisa diterapkan adalah dengan menambah jumlah *buffer* dan mengubah skema pengiriman ulang sms-nya (konfigurasi ulang). Setelah dilakukan perubahan jumlah *buffer* (menjadi 150.000 sms) dan skema pengiriman ulang sms untuk SDC 83800 (menjadi 818 *retry scheme*) di sisi konfigurasi SMSC, maka dilakukan analisa atau

pengecekan kembali terhadap performansi trafik SMS A2P SDC 83800.



Gambar 4.3 Grafik Performansi Trafik SMS setelah perubahan jumlah *buffer*

Dari gambar 4.3, dapat dilihat bahwa *error code* 65 untuk trafik SMS A2P SDC 83800 mulai hilang setelah implementasi. Dan hanya terlihat *error code* 0 (sms sukses terkirim ke pelanggan). Dengan demikian performansi trafik SMS A2P SDC 83800 kembali normal. Berikut *sample log*-nya,

```
<382
CBTSMSC1:/home/smc/billdata/bpsbill/11/ba
ckup>CdrReaderMO.sh

put CDR below, then hit <<enter>>:

503460947,86800,6283109920170,,62818445009,0,0,0,
0,0,0,0,0,,0,153,0,0,0,2014/05/21
10:35:12,,,0,soa,,,,,,,,,,,,,Gateway_G,,,,,,,,,
""
Originator Addr      : 83800
Destination Addr     : 6283109920170
Result                : 0
Original MO Submit Time :      2018/05/21
10:35:12
```

Gambar 4.4 *command* dan *log* hasil penarikan data pada server

Dari *sample log* SMSC gambar 4.4, dapat dilihat bahwa SMS A2P dari SDC 83800 yang dikirim ke 6283109920170 sukses terkirim ke penerima (*Result* 0).

Selain menganalisa penyebab penuhnya *buffer*, dan performansi akan dianalisa juga hal-hal yang diperbaiki oleh metode penambahan *buffer* dan perubahan skema pengiriman ulang sms di sisi konfigurasi SMSC, diantaranya meningkatnya performansi trafik SMS A2P SDC 83800 tanpa mengurangi jumlah pengiriman trafik dari sisi aplikasi dan tetap menggunakan SMSC yang sama, kemudian hilangnya *error code 65 "throttled for A#"* (indikasi penuhnya *buffer* SMSC), SMSC dapat menampung trafik SMS A2P SDC 83800 lebih banyak, dan dengan diubahnya skema pengiriman ulang SMS A2P SDC 83800 maka sms yang ditampung tidak akan disimpan terlalu lama di *buffer*.

V. Simpulan

Dari hasil pengujian dan analisa dapat disimpulkan sebagai berikut :

1. Hal-hal yang menyebabkan penuhnya *buffer* di SMSC untuk trafik SMS A2P SOA SDC 83800 ialah meningkatnya jumlah trafik yang dikirim *user* SOA dari biasanya (mencapai 3 kali lipat, dikarenakan adanya penambahan data pelanggan), naiknya *error absent subscriber, subscriber power off* dan *subscriber busy* di sisi FDA (ketiga error ini merupakan *error* dari sisi pelanggan yang biasa disebut *user behavior error*).
2. Dengan metode penambahan *buffer* (menjadi 150.000 sms) dan mengubah skema pengiriman ulang SMS A2P SDC 83800 (menjadi 818 *retry scheme*) di sisi konfigurasi SMSC, tanpa mengurangi jumlah trafik yang dikirim dari sisi aplikasi dan tetap menggunakan SMSC yang sama, terlihat *error code 65 "Throttled for A#"* yang mengalami kenaikan pada performansi trafik SMS A2P SOA SDC 83800. Sehingga metode penambahan *buffer* dan perubahan skema pengiriman ulang sms dapat dijadikan salah satu solusi untuk menghilangkan *error 65 "throttled for A#"* dan untuk meningkatkan performansi trafik SMS A2P SDC 83800.

Daftar Pustaka

- [1] Bhatia, Ashish. "Mobile Originated (MO) SMS Flow".
[http://www.TelecomTigersMobileOriginated\(MO\)SMSFlow.html](http://www.TelecomTigersMobileOriginated(MO)SMSFlow.html). 2009
- [2] Feri, Muhammad. "Analisis Antrian untuk SMSC pada GSM di PT. Telkomsel". Tugas Akhir Sarjana Teknik Universitas Sumatra Utara. 2007
- [3] Gupta, Punet. "Definisi SMS". 2000
- [4] Ibrahim, Ali. "Pengembangan Sistem Informasi Monitoring Tugas Akhir Berbasis Short Message Service (SMS) Gateway di Fasilkom Unsri". 2011
- [5] Irwan. "Proses Pengiriman SMS".
<http://www.ProsesPengirimanSMS-silveriusirwan.htm>. 2011
- [6] Nasir, Muhamad. "Sistem Informasi Perpustakaan Online Berbasis PHP Dan SMS Gateway Di Perpustakaan Umum Bengkalis". Tugas Akhir Sarjana Teknik Informatika Politeknik Elektronika Negeri Surabaya Institut Teknologi Sepuluh Nopember. 2003
- [7] Netsize. "Tiga Jenis Encoding untuk menerjemahkan Pesan pada SMS". 2003
- [8] Rozidi, Imron. "Membuat Sendiri SMS Gateway (ESME) Berbasis Protokol SMPP". Penerbit Andi. Yogyakarta. 2004, hal: 1
- [9] Setemen, Komang. "PENGEMBANGAN SISTEM INFORMASI PEMANTAUAN KULIAH BERBASIS SMS DI JURUSAN MANAJEMEN INFORMATIKA". Staf Pengajar pada Jurusan Manajemen Informatika FTK Undiksha. ISSN: 1829-5282, 2010
- [10] Suhaidi, Mustazzihim. "Analisis Dan Perancangan Sistem Informasi Pengisian Pulsa Elektronik Berbasis SMS (Studi Kasus 4VR1.CELL Magelang)". Tugas Akhir Strata 1 Teknik Informatika AMIKOM Yogyakarta. 2010
- [11] Utomo, Prasetya Ambang. "Membangun Aplikasi SMS Berbasis Open Source" Penerbit Andi. Yogyakarta. 2006
- [12] Wikipedia. "Short message service technical realization (GSM)".
[http://www.Shortmessageservicetechnicalrealization\(GSM\)-Wikipedia.thefreencyclopedia.htm](http://www.Shortmessageservicetechnicalrealization(GSM)-Wikipedia.thefreencyclopedia.htm). 2014
- [13] Wisnu, Rahma. "Data Structure - Bab 4.pdf". 2006