JURNAL TEKNOLOGI TECHNOSCIENTIA ISSN: 1979-8415
Vol. 13 No. 1 Agustus 2020 E-ISSN: 2714-8025

ANALISIS KEAMANAN DATA SELULER TERHADAP SERANGAN SNIFFING MENGGUNAKAN RTL-SDR

Yulivia Rhadita Savitri[1], Sopian Soim[2], Mohammad Fadhli[3]
[1,2,3]Politeknik Negeri Sriwijaya
Email: [1]yulivia2299@gmail.com, [2]sopian_soim2005@yahoo.com,
[3]mohammad.fadhli@polsri.ac.id

Masuk: 19 Juli 2020, Revisi masuk: 09 Agustus 2020, Diterima: 10 Agustus 2020

ABSTRACT

Data security on cellular networks is needed to maintain privacy and avoid illegal actions, because the process of communication and sending data remotely using cellular networks, especially GSM and GPRS are still widely used than conventional methods. One of the potential for sniffing data security holes can occur in the air interface, it is signalling system when transmitting data between user devices or Mobile Station (MS) and Base Transceiver Station (BTS). Therefore, to find out the quality of data security on GSM and GPRS cellular networks, a penetration testing method with RTL-SDR device are tested. The results obtained are the cellular network security system used is still secure based on data transmission patterns using timeslot, Temporary Mobile Subscriber Identity (TMSI), and changes in GSM Frame Number on GSM. Then, network security on GPRS is also still safe using logic channels, timeslot, and different security algorithm from GSM.

Keywords: *Air interface, BTS, Cellular, RTL-SDR, Security*.

INTISARI

Keamanan data pada jaringan seluler sangat diperlukan untuk menjaga privasi dan menghindari tindakan ilegal, karena proses komunikasi dan pengiriman data jarak jauh menggunakan jaringan seluler khususnya GSM dan GPRS masih banyak digunakan dibandingkan cara konvensional. Salah satu potensi terbukanya celah keamanan data dapat terjadi pada *air interface*, yaitu pada saat transmisi data antara perangkat *user* atau *Mobile Station* (MS) dan *Base Transceiver Station* (BTS). Maka, untuk mengetahui kualitas keamanan data pada jaringan seluler GSM dan GPRS dilakukan pengujian dengan metode *penetration testing* dengan perangkat RTL-SDR. Hasil yang diperoleh adalah sistem keamanan jaringan seluler yang digunakan masih aman berdasarkan pola transmisi data yang menggunakan *timeslot*, *Temporary Mobile Subscriber Identity* (TMSI), dan perubahan *GSM Frame Number* pada GSM. Kemudian, keamanan jaringan pada GPRS juga masih aman menggunakan kanal logika, *timeslot*, dan algoritma keamanan yang berbeda dari GSM.

Kata-kata kunci: Air interface, BTS, Keamanan, RTL-SDR, Seluler.

PENDAHULUAN

Sistem telekomunikasi seluler masih menjadi sistem utama dalam proses komunikasi jarak jauh, karena lebih efisien dalam proses pengiriman maupun biaya. Untuk menjaga privasi komunikasi diperlukan keamanan data pada jaringan seluler. Keamanan data adalah ilmu pengetahuan dan pembelajaran mengenai metode perlindungan data pada komputer dan sistem komunikasi. Salah satu potensi terbukanya celah keamanan data dapat terjadi saat sebuah data ditransmisikan dari satu perangkat *user* ke perangkat *user* lain melalui sebuah jaringan, baik jaringan dengan transmisi *wired* maupun *wireless.*

Data yang dikirimkan dapat melalui sebuah media yang terhubung ke jaringan, seperti data pada SMS yang dikirimkan antara *Mobile Station* melalui *air interface* dan *email* yang dikirimkan ke tujuan dengan sistem Postfix dengan fungsi sebagai *Mail Transfer Agent* (MTA), yaitu pengelola dan penyalur *email* dari *user* hingga masuk ke jaringan. SMS dan *email* yang dikirimkan ke jaringan berupa sinyal yang terenkripsi. Dalam suatu jaringan, sinyal tersebut ditransmisikan menuju BTS seluler yang telah ditentukan hingga SMS dan *email* tersebut sampai ke perangkat (*Mobile Station*) penerima. Namun, saat sinyal ditransmisikan menuju BTS, sinyal tersebut dapat ditangkap oleh *sniffer* untuk

mendapatkan informasi secara ilegal, sehingga terjadi kasus *sniffing*.

RTL-SDR bekerja pada frekuensi 24 MHz hingga 1766 MHz dan berfungsi sebagai *receiver* (Rx) (Laufer, 2014). Saat *sniffing* berjalan, *sniffer* akan mencari frekuensi jaringan yang digunakan *user* dalam proses pengiriman SMS dan *email*. Setelah frekuensi yang tepat ditemukan, *sniffer* mulai menangkap sinyal yang dilewatkan pada frekuensi tersebut menggunakan RTL-SDR. RTL-SDR didukung oleh *tools* pada sistem operasi berbasis Linux yang dapat me-*capture* sinyal tangkapan RTL-SDR. Setelah sinyal berhasil di-*capture, tools* pada Linux akan melakukan proses *penetration testing* sinyal dengan me-*decode* atau mendekripsi sinyal hingga informasi data yang dikirim *user* dapat diketahui oleh *sniffer.*

Rumusan masalah pada penelitian ini adalah:

1. Bagaimana proses pengujian penyerangan data melalui *sniffing*?
2. Metode apa yang digunakan untuk mengamankan data dan seberapa akurat data tersebut dapat diamankan?
3. Bagaimana cara mengukur kinerja suatu proses keamanan data?

Adapun tujuan yang akan dicapai pada penelitian ini adalah:

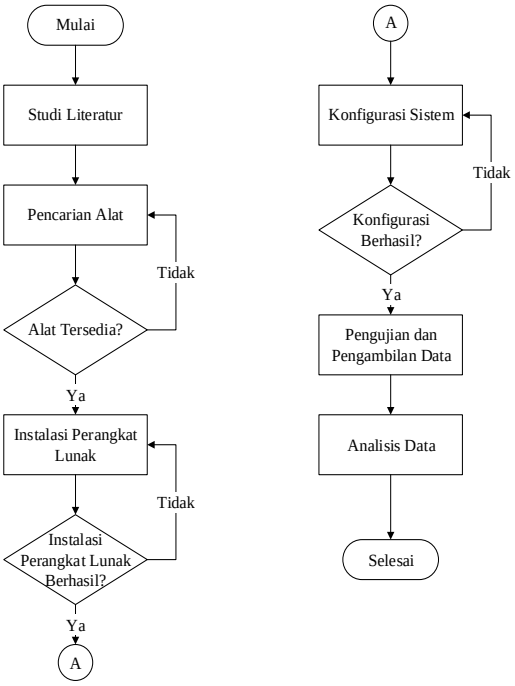
1. Untuk mengetahui tingkat kesuksesan keamanan data pada jaringan seluler.
2. Sebagai bahan analisis terhadap informasi sinyal yang ditangkap dari perangkat RTL-SDR.
3. Untuk mengetahui perbandingan teknik enkripsi berdasarkan media pengiriman data (SMS dan *email*).

METODE

Proses pengujian dilakukan di *coverage area* BTS (*Base Transceiver Station*) *provider* Telkomsel dengan frekuensi pada *band* 900 MHz di Kota Palembang. Gambar 1 menampilkan alur penelitian yang dilakukan.

Deskripsi alur penelitian:

1. Studi literatur yang dilakukan pada penelitian ini adalah pencarian referensi yang terkait dengan pembahasan mengenai sistem keamanan data, proses pengiriman data, ancaman keamanan data sistem *wireless*, dan teknologi jaringan seluler. Selain itu, pada tahapan ini juga dilakukan analisis perbandingan terhadap penelitian yang telah dipublikasikan sebelumnya dari berbagai jurnal, prosiding, dan sumber pustaka lainnya.
2. Proses pencarian alat pada penelitian ini berupa penyediaan perangkat RTL-SDR dan penyesuaian spesifikasi perangkat pendukung penelitian.
3. Instalasi perangkat lunak yang dilakukan adalah instalasi sistem operasi Kali Linux 2018, Linux Ubuntu 18.04 LTS, dan Ubuntu 16.04 LTS di dalam *software* Oracle VM VirtualBox, sehingga Kali Linux, Linux Ubuntu 18.04 LTS, dan Ubuntu 16.04 LTS akan bekerja pada mode virtual.
4. Konfigurasi sistem pada *mail server* Postfix dan integrasi *mail server* Postfix dan Gmail.
5. Pengujian pengiriman data antar *user* yang berisi teks melalui SMS dan *email*.
6. Pengambilan data melalui proses penangkapan sinyal menggunakan RTL-SDR dan *tools* pada sistem operasi Kali Linux 2018 dan Linux Ubuntu 18.04 LTS yang berfungsi me-*decode* sinyal ke dalam bentuk paket data.
7. Analisis data yang diperoleh dilakukan berdasarkan kemampuan sinyal yang terenkripsi dapat di-*decode* pada SMS dan *email*.



Gambar 1. Alur Penelitian

Pengujian *sniffing* data seluler menggunakan beberapa perangkat keras dan perangkat lunak seperti ditampilkan pada Tabel 1 dan Tabel 2.

Tabel 1. Kebutuhan Perangkat Keras

No	Perangkat Keras	Spesifikasi
1	RTL-SDR	R2832U USB 2.0 24-1766 MHz *Receiver*
2	Laptop Asus X45A	Intel Celeron 1000M 1.80 GHz RAM 4 GB 64 bit HDD 1 TB USB 2.0 & 3.0
3	*Smartphone* Samsung Galaxy V	4.4 Kitkat RAM 512 MB ROM 4 GB CPU 1.2 GHz *Dual Core* *Micro* SIM

Tabel 2. Kebutuhan Perangkat Lunak

No	Perangkat Lunak	Fungsi
1	Oracle VM VirtualBox	*Software* Sistem Operasi Virtual
2	Kali Linux 2018	Sistem Operasi
3	Linux Ubuntu 18.04 LTS	Sistem Operasi
4	Linux Ubuntu 16.04 LTS	Sistem Operasi
5	GQRX	SDR *Tool*
6	Gr-GSM	SDR *Tool*
7	Kalibrate	*Scanning* BTS
8	Frequency Check	Pemindai ARFCN
9	Airprobe	*Decoding Tool*
10	GSM Framecoder	*Decoding Tool*
11	Wireshark	*Capture Packet Data*

Pengambilan data dilakukan dengan metode *penetration testing*, yaitu metode pengujian penyadapan (*sniffing*) terhadap target yang menerima data berupa SMS dan *email* yang telah dirancang. Proses pengambilan data dimulai pada saat perangkat pengirim telah mengirimkan SMS dan *email* dalam bentuk teks menuju target, maka RTL-SDR menangkap sinyal pada *air interface* BTS (*downlink*) berdasarkan ARFCN yang digunakan perangkat target. Selanjutnya, dilakukan proses dekripsi (*decoding*) sinyal untuk menemukan SMS dan *email* target.

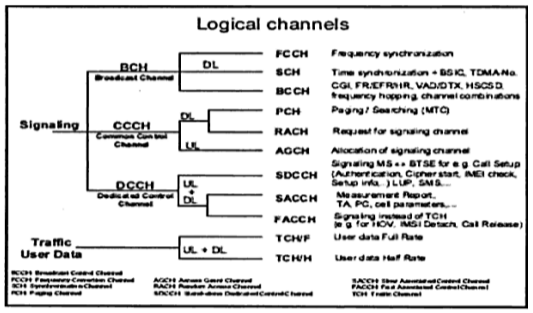
Tahapan penyadapan SMS dan *email* dilakukan dengan cara yang sama, namun dengan parameter yang berbeda. Keberhasilan atau kegagalan penemuan data SMS dan *email* menjadi penentu kualitas keamanan jaringan GSM seluler.

Parameter tingkat keamanan data yang diukur pada pengujian penyadapan SMS dan *email* adalah sebagai berikut.

Tabel 3. Parameter Keamanan Data

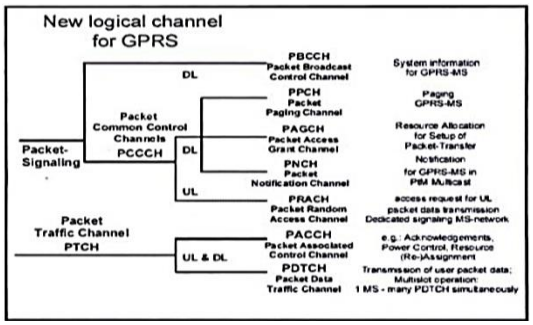
No	SMS	Email
1	ARFCN	ARFCN
2	*Broadcast Channel*	*Location Area Identification*
3	*Dedicated Control Channel*	*Packet Broadcast Control Channel*
4	*Timeslot*	GPRS *Indicator*
5	*Location Area Identification*	*Packet Dedicated Control Channel*
6	TMSI	*Timeslot*
7	*GSM Frame Number*	*Ciphering Mode Command*
8	*Ciphering Mode Command*	
9	*Burst frame*	

GSM (*Global System for Mobile*) adalah generasi kedua dari standar sistem seluler. Teknologi GSM (Gambar 2) menggunakan sistem TDMA dengan alokasi kurang lebih sekitar delapan pengguna di dalam satu *channel* frekuensi sebesar 200 kHz per satuan waktu (Apriyanti, dkk., 2016).



Gambar 2. Kanal Logika GSM
(Wardhana dan Makodian, 2010)

Kanal logika GPRS (Gambar 3) dibagi menjadi tiga, yaitu PTCH (*Packet Traffic Channel*), PCCH (*Packet Common Control Channel*), dan PBCCH (*Packet Broadcast Control Channel*) (Wardhana dan Makodian, 2010).



Gambar 3. Kanal Logika
(Wardhana dan Makodian, 2010)

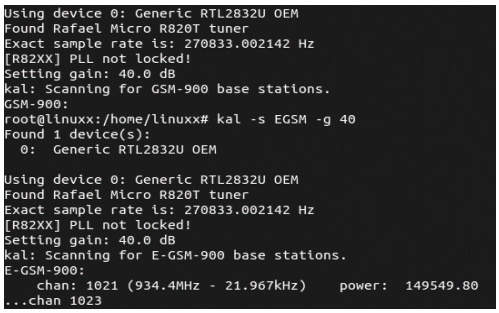
Decoding adalah proses konversi data yang telah dikirimkan oleh sumber pesan menjadi informasi yang dimengerti oleh penerima (Rivaldy, dkk., 2017). *Software Defined Radio* dapat menerjemahkan sinyal yang ditangkap oleh perangkat keras berupa *transmitter* atau *receiver* kemudian diterjemahkan sebagai proses *decoding* sinyal itu sendiri (Ramadhan, dkk., 2018).

PEMBAHASAN

Identifikasi *Channel*

Identifikasi *channel* bertujuan untuk mengetahui lokasi *channel* GSM yang digunakan *smartphone* atau perangkat target dan untuk mengidentifikasi *channel* GSM yang tersedia di area pengujian. Tahapan pertama dari identifikasi *channel* yaitu menggunakan *tools* Kalibrate yang telah diinstalasi pada Kali Linux 2018 dan Linux Ubuntu 18.04 LTS. Selain itu, Kalibrate juga berfungsi untuk mengkalibrasi nilai *offset* RTL-SDR dan mengetahui apakah RTL-SDR diatur pada posisi frekuensi yang tepat atau dengan nilai *offset* tertentu.

Pada Gambar 4 adalah hasil identifikasi *channel* yang diterima oleh RTL-SDR. Pada bagian pertama adalah hasil *scanning* pada *channel* GSM900 dimana pada *band* tersebut tidak ditemukan *channel* GSM utama atau *channel* dengan *power* terbesar di area pengujian. Sedangkan, pada bagian kedua adalah hasil *scanning* pada *channel* EGSM dengan *band* 900 MHz dimana terdapat *channel* utama dan terbesar dengan nomor *channel* 1021, frekuensi 934.4 MHz, nilai *offset* sebesar 21.967 kHz, dan *power* sebesar 149549.80.

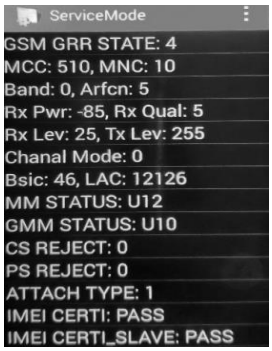


Gambar 4. Identifikasi *Channel Band* 900 MHz

Identifikasi *Channel* Perangkat Target

Perangkat target (*smartphone*) diatur ke dalam *Service Mode* dengan me-*dial* nomor *#0011#. Selanjutnya, diperoleh informasi mengenai *channel* yang digunakan perangkat target. Informasi tersebut diantaranya adalah MCC (*Mobile Country Code*) atau kode area *provider* yaitu 510 yang menunjukkan kode Negara Indonesia, MNC (*Mobile Network Code*) atau kode *provider* Telkomsel yaitu 10, Band 0 merupakan kode *band* frekuensi yang digunakan perangkat yaitu pada *band* 900 MHz, dan informasi utama yang diperlukan dalam mendukung *sniffing* data adalah ARFCN (*Absolute Radio Frequency Channel Number*) atau nomor *channel* yang digunakan perangkat.

Kedudukan ARFCN perangkat bersifat tidak tetap, perpindahan ARFCN tergantung pada kekuatan sinyal yang dipancarkan BTS pada tiap *channel*. Apabila kekuatan sinyal yang digunakan mulai lemah, maka perangkat akan mencari *channel* dengan kekuatan yang lebih optimal di sekitarnya. Pada Gambar 5 menunjukkan ARFCN perangkat dengan nomor 5 pada *provider* Telkomsel yang paling sering digunakan perangkat target. *Power* sinyal yang diterima perangkat target dari *channel* sebesar -85 dBm, level sinyal yang diterima sebesar -25 dBm, kualitas sinyal yang diterima berskala 5, dan level sinyal yang dipancarkan oleh BTS sebesar 255 dBm.



Gambar 5. Informasi BTS yang Digunakan *Smartphone*

Identifikasi Frekuensi *Downlink Channel*

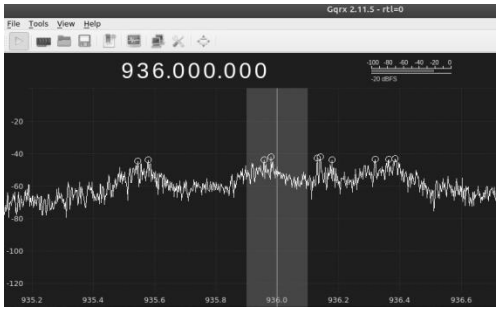
Identifikasi frekuensi yang digunakan *smartphone* menggunakan *website* Frequency Calculator dengan cara memasukkan nomor ARFCN perangkat dan mengatur *network type* menjadi GSM. Maka, pada Gambar 6 diperoleh frekuensi *uplink* sebesar 891 MHz, frekuensi *downlink* sebesar 936 MHz, dan *bandwidth channel* sebesar 0.2 MHz atau 200 kHz.

Result

Network Type	GSM (TDMA)
E/U/ARFCN	5
Band Name	GSM 900
Uplink Frequency (phone to base station)	891 MHz
Downlink Frequency (base station to phone)	936 MHz
Band Number	900
Possible Bandwidths	0.2 MHz
Sector Color	

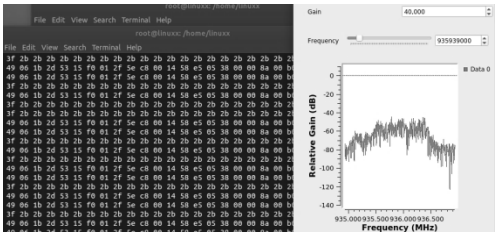
Gambar 6. Frekuensi dan *Bandwidth* ARFCN 5

Frekuensi *downlink* yang diperoleh selanjutnya dilakukan pencarian frekuensi yang tepat dan dengan *power* terbesar menggunakan GQRX dimana *range* frekuensi nomor *channel* 5 adalah antara 935.9 MHz sampai 936.1 MHz Gambar 7. Hasil yang diperoleh adalah terdapat dua posisi frekuensi dengan *power* terbesar pada ARFCN 5 yaitu frekuensi 935.939 MHz dan frekuensi 935.975 MHz. *Power* sinyal yang kuat menandakan bahwa terdapat informasi yang ditransmisikan pada frekuensi tersebut. Selain itu, frekuensi dengan *power* terbesar menunjukkan potensi dari posisi frekuensi yang digunakan perangkat.



Gambar 7. Frekuensi *Downlink* ARFCN 5

Kemudian, frekuensi yang diperoleh dengan *power* terbesar diuji coba pada Gr-GSM untuk mengetahui apakah sinyal dapat diterima oleh RTL-SDR dan dapat di-*decoding*. Untuk menjalankan Gr-GSM menggunakan *command* "grgsm_livemon", kemudian Gain diatur sebesar 40 dB dan frekuensi dengan *power* terbesar diatur ke dalam Gr-GSM. Kode-kode heksadesimal yang ditampilkan pada terminal menunjukkan bahwa terdapat informasi sinyal yang berhasil di-*decode* seperti yang ditunjukkan pada Gambar 8. Selama proses *decoding* terjadi *error decode* yang ditandai dengan terhambatnya proses *decoding* akibat dari RTL-SDR yang tidak dapat menerima sinyal secara signifikan.



Gambar 8. Tangkapan Sinyal pada ARFCN 5

Pengujian dan Analisis

Proses pengujian *sniffing* data menggunakan metode *penetration testing* yang didukung oleh *tools*, perangkat RTL-SDR, dan *smartphone*. Metode *Penetration testing* adalah metode simulasi penyerangan terhadap jaringan ataupun sistem jaringan yang diuji untuk mengetahui kualitas keamanan jaringan tersebut. Pengujian dilakukan menggunakan dua target data yaitu SMS dan *email*. Kemudian, analisis yang dilakukan berdasarkan hasil tangkapan sinyal menggunakan RTL-SDR, proses *decoding* dan informasi yang diperoleh setelah *decoding* sinyal yang selanjutnya dihubungkan dengan parameter-parameter keamanan yang telah ditentukan.

Pengujian *Sniffing* SMS

Sniffing SMS menggunakan *tools* Gr-GSM dengan me-*capture* sinyal pada ARFCN 5. Pola pengujian yang dilakukan pada *sniffing* SMS adalah perangkat pertama melakukan pengiriman pesan berisi teks melalui SMS secara berurutan kepada target, kemudian *sniffer* melakukan penangkapan sinyal menggunakan RTL-SDR berdasarkan identifikasi *channel* yang telah dilakukan pada perangkat target. Untuk menjalankan penangkapan sinyal dilakukan dengan *command* "grgsm_capture –f 5935939000 –s 1e6 –c tsel93.cfile –g 40" seperti yang ditunjukkan pada Gambar 9 "grgsm_capture" menunjukkan perintah *capture* sinyal menggunakan Gr-GSM. Kode "-f 935939000" menunjukkan frekuensi dengan ARFCN 5. Kode "-s 1e6" menunjukkan nilai *sample rate* saat *capturing* sinyal yaitu sebesar 1 MHz. Kode "-c tsel93.cfile" adalah hasil tangkapan sinyal dari RTL-SDR yang disimpan dalam ekstensi *.cfile. Kode "-g 40" merupakan

gain yang digunakan sebesar 40 dB pada saat penangkapan sinyal.

```
root@linuxx:/home/linuxx# grgsm_capture -f 935939000 -s 1e6 -c tsel93.cfile -g 40
```

Gambar 9. *Capturing* Sinyal

Selanjutnya, tahapan *decoding* sinyal (Gambar 10) yang tersimpan dalam ekstensi *.cfile agar informasi dapat ditampilkan ke dalam Wireshark. Untuk melakukan *decoding* sinyal dapat menuliskan *command* "grgsm_decode –f 935939000 –c tsel93.cfile –s 1e6 –m BCCH –t 0". Kode "-m BCCH" menunjukkan *Broadcast Channel* (BCH) yang akan di-*decode* adalah BCCH (*Broadcast Control Channel*) dan "-t 0" yang menampilkan *timeslot* 0.

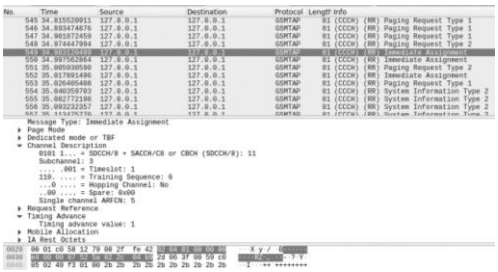
BCCH akan menampilkan informasi mengenai BTS (*Base Transceiver Station*) yang digunakan MS (*Mobile Station*) atau perangkat target. Informasi yang akan ditampilkan dari BCCH antara lain DCCH (*Dedicated Control Channel*), *timeslot*, *hopping channel*, ARFCN (*Absolute Radio Frequency Channel Number*), *timing advance*, dan LAI (*Location Area Identification*).

```
root@linuxx:/home/linuxx# grgsm_decode -f 935939000 -c tsel93.cfile -s 1e6 -m BCCH
-t 0
```

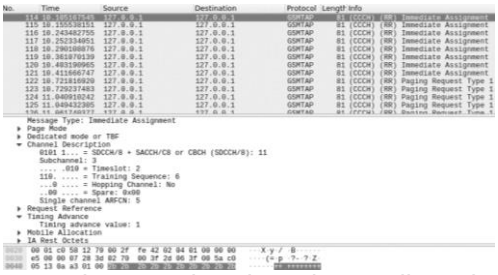
Gambar 10. *Decoding* Sinyal

Kemudian, Wireshark menampilkan hasil *decoding* sinyal yang telah ditangkap menggunakan RTL-SDR seperti pada Gambar 11 dan Gambar 12. Pada *Immediate Assignment* terdapat *Channel Description* dimana terdapat informasi jenis *Dedicated Control Channel* (DCCH) yaitu menggunakan SDCCH/8. SDCCH (*Stand Alone Dedicated Control Channel*) adalah *bi-directional channel* yang berarti kanal dengan dua arah pancaran sinyal. SDCCH digunakan dalam sistem pensinyalan, *call setup*, autentikasi, *location update*, trafik kanal, transmisi pesan singkat dari atau menuju MS, dan transmisi data antara MS dan *network*. Konfigurasi SDCCH/8 yang digunakan BTS menunjukkan bahwa delapan *sub channels* yang digunakan untuk pensinyalan yang diposisikan pada 1 TS (*Timeslot*) hingga menghasilkan satu trafik kanal lebih sedikit pada *cell*.

Pada *Immediate Assignment* juga ditemukan *timeslot* saat pengiriman data. *Timeslot* adalah periode waktu diskrit secara *real time* dimana sebuah data harus sampai agar dapat di-*decode* penerima. *Timeslot* dibagi menjadi beberapa alokasi saat transmisi data dilakukan, sehingga pada *frame* tertentu data akan berada pada *timeslot* yang berbeda. Pada Gambar 11 dan Gambar 12 diperoleh dua buah *timeslot* yang digunakan pada transmisi data, yaitu *timeslot* 1 dan *timeslot* 2. Pembagian tersebut menunjukkan bahwa data GSM yang dikirimkan dibagi menjadi beberapa slot untuk menjaga keamanan data GSM, sehingga mempersulit *sniffer* dalam menemukan data.



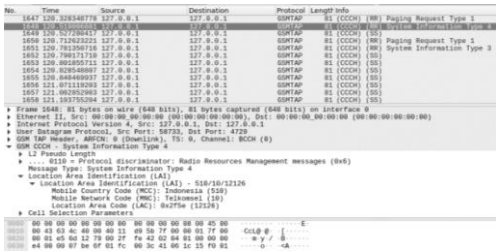
Gambar 11. *Timeslot* 1 dan *Dedicated Control Channel* SDCCH/8



Gambar 12. *Timeslot* 2 dan *Dedicated Control Channel* SDCCH/8

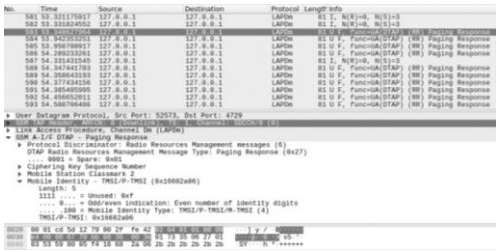
Kemudian, terdapat keterangan *hopping channel* dengan status "No" yang artinya MS sedang tidak melakukan panggilan atau *dial*. Nomor ARFCN 5 yang ditemukan pada *Immediate Assignment* sesuai dengan penggunaan perangkat target. Selanjutnya, *timing advance* menunjukkan jarak antara MS dan BTS. Informasi yang diperoleh menunjukkan *timing advance* bernilai 1, artinya jarak MS dan BTS antara 553,5 hingga 1107 meter.

Pada *System Information Type* 4 (Gambar 13) diperoleh informasi berupa LAI (*Location Area Identification*) yang menunjukkan bahwa sinyal yang ditangkap RTL-SDR adalah sinyal GSM yang berasal dari Negara Indonesia dengan kode 510, operator seluler dari Telkomsel dengan kode 10, dan *Location Area Code* yang ditangkap dengan kode 12126. Informasi tersebut menunjukkan persamaan informasi yang diperoleh dari perangkat target.



Gambar 13. Informasi pada *System Information Type* 4

Selanjutnya adalah menemukan TMSI, *GSM Frame Number*, dan algoritma keamanan GSM *provider* Telkomsel. *Decoding* dimulai dari SDCCH/8 *timeslot* 1 dengan *command* "grgsm_decode –f 935939000 –s 1e6 –c tsel93.cfile –m SDCCH8 –t 1" pada terminal. Pada *timeslot* 1 ditemukan *packet list* berupa *Paging Response* seperti yang ditunjukkan pada Gambar 14.

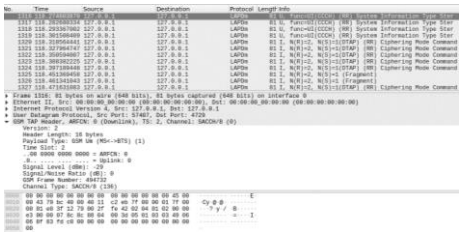


Gambar 14. TMSI pada *Paging Response*

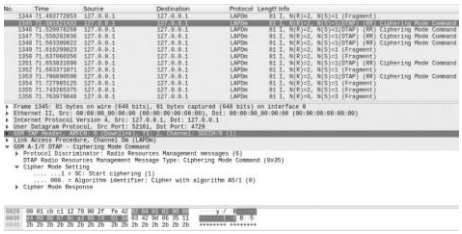
Pada *Paging Response* ditemukan TMSI dengan kode yang sama pada tiap *packet list* yaitu 0x16682a06. TMSI merupakan nomor IMSI sementara yang bersifat acak, sehingga *sniffer* tidak mengetahui nomor IMSI perangkat. TMSI dengan nomor yang sama tersebut mempunyai indikasi bahwa informasi yang diperoleh adalah informasi perangkat target berdasarkan pola pengiriman SMS oleh target yang dilakukan secara berurutan.

Informasi algoritma keamanan data GSM tidak ditemukan pada *timeslot* 1, artinya informasi tersebut memungkinkan tersedia pada *timeslot* 2. Untuk me-*decode timeslot* 2 dilakukan dengan *command* "grgsm_decode –f 935939000 –s 1e6 –c tsel93.cfile –m SDCCH8 –t 2".

Pada Gambar 15 dan Gambar 16 terdapat *System Information Type 5ter* dan *Ciphering Mode Command* yang merupakan hasil dari *decoding timeslot* 2. Pada *System Information Type 5ter* berisi informasi *GSM Frame Number* yaitu dengan nomor 494732. *GSM Frame Number* juga mempunyai nomor yang selalu berubah setiap *frame*. *GSM Frame number* diperlukan pada saat *burst frame* dilakukan. Pada umumnya perubahan *GSM Frame Number* pada saat pengiriman data terjadi setiap 102 *frame*. Jika *GSM Frame Number* adalah 494732, maka 102 *frame* berikutnya adalah 494834 dan 102 *frame* berikutnya lagi adalah 494936.



Gambar 15. *GSM Frame Number* pada *System Information Type 5ter*

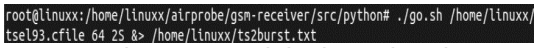


Gambar 16. Algoritma A5/1 sebagai Pengaman Data

Kemudian, pada *packet list Ciphering Mode Command* menunjukkan algoritma keamanan yang digunakan *provider* Telkomsel yaitu Algoritma A5/1. Algoritma A5/1 digunakan untuk mengenkripsi data yang ditransmisikan antara MS dan BTS pada *air interface* (media komunikasi transmisi data melalui frekuensi radio).

Setelah menemukan *GSM Frame Number*, TMSI, dan algoritma keamanan GSM. Selanjutnya adalah proses *burst* yang bertujuan untuk mencari KC (*Key Ciphering*) perangkat target. Tahapan yang dilakukan yaitu masuk ke direktori *tool* Airprobe dengan *command* "cd airprobe/gsm-receiver/src/python", kemudian masukkan *command* "./go.sh /home/linuxx/tsel93.cfile 64 2S &> /home/linuxx/ts2burst.txt".

Kode "./go.sh" (Gambar 17) adalah perintah untuk menjalankan *file* gsm-receiver.py yang berfungsi memfilter bit-bit informasi hasil tangkapan sinyal *file* ekstensi *.cfile."/home/linuxx/tsel93.cfile" menunjukkan lokasi dan nama penyimpanan *.cfile yang akan di-*burst*. "64" menunjukkan *decimation rate*, "2S" adalah *timeslot* 2, dan "&> /home/linuxx/ts2burst.txt" adalah tujuan penyimpanan *burst* dan nama *burst*.

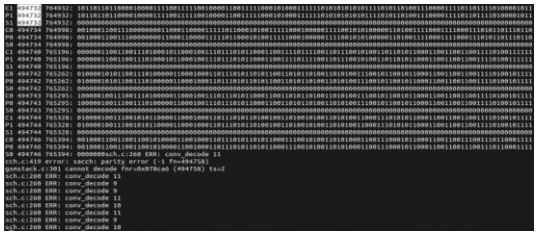


Gambar 17. Menjalankan Airprobe

Pada Gambar 18 menunjukkan hasil *burst* menggunakan Airprobe. *Burst* adalah urutan bit yang ditransmisikan oleh BTS atau MS. Untuk menemukan KC, terlebih dahulu dilakukan pencarian *burst frame* berdasarkan *GSM Frame Number* yang telah diperoleh dari *System Information Type 5ter*. *GSM Frame Number* tersebut adalah 494732 dengan menuliskan perintah "/494732".

Gambar 18. Hasil *Burst* .cfile

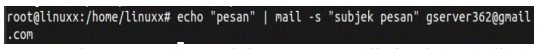
Pada Gambar 19 menunjukkan penemuan *GSM Frame Number* pada *burst file* tersebut. Kode C menunjukkan bit-bit *burst* yang terenkripsi, kemudian kode P menunjukkan bit-bit *burst* yang terdekripsi, lalu kode S menunjukkan bit-bit *keystream*, yaitu bit-bit terenkripsi XOR dengan bit-bit terdekripsi. Hasil *burst* tersebut tidak dapat didekripsi secara langsung untuk mendapatkan data, karena bit-bit terdekripsi pada *burst* sama dengan bit-bit *burst* yang terenkripsi. Jika kode C, P, dan S mempunyai nilai 1 di belakangnya, menunjukkan bahwa bagian tersebut adalah *burst frame* pertama. Kemudian, untuk nilai kedua merupakan nomor *frame*, nomor ketiga adalah hasil modifikasi atau perubahan nomor *frame* berdasarkan ketentuan dari algoritma A5/1.


Gambar 19. *GSM Frame Number* 494732

Untuk mencari KC (*Key Ciphering*) dilakukan menggunakan *tool* Kraken dengan memasukkan bit-bit data *burst* yang terletak pada *burst file* setelah *burst frame* terakhir. Namun, bit-bit data *burst* tersebut tidak berhasil ditemukan akibat terjadinya *error decode* yang menyebabkan kesalahan *decode* bit selama proses *burst* dilakukan. Kesalahan *burst* terjadi karena adanya distorsi pada saat penangkapan sinyal secara *live capture* akibat transmisi data pada *air interface* berlangsung cepat, sehingga terdapat bit-bit yang tidak berhasil di-*decode* oleh RTL-SDR.

Pengujian *Sniffing* Email

Pengujian *sniffing email* dilakukan dengan pola pengiriman *email* berisi teks dari *email* Postfix pada Linux Ubuntu 16.04 LTS menuju Gmail yang telah disinkronkan pada perangkat atau *smartphone* target. Kemudian, penerima *email* atau target mengakses Gmail pada perangkat dan membuka *email* yang telah diterima dari sistem Postfix di jaringan 2G. Pada saat target mengakses Gmail, maka proses *sniffing* data dilakukan menggunakan RTL-SDR dengan cara menangkap sinyal yang digunakan perangkat target dilanjutkan dengan proses *decoding* sinyal untuk menemukan informasi atau *email* target. Pengiriman *email* ke akun Gmail target dilakukan dengan menuliskan "echo "pesan" | mail –s "subjek pesan" gserver362@gmail.com seperti pada Gambar 20.



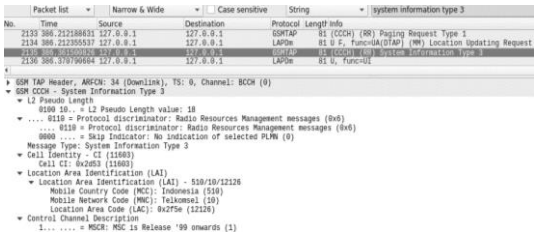
Gambar 20. Pengiriman *Email* dari Postfix

Kemudian, *email* berhasil diterima oleh target pada akun Gmail seperti yang ditunjukkan pada Gambar 21. Pada saat pengiriman *email* dari Postfix, dilakukan penangkapan sinyal menggunakan RTL-SDR pada frekuensi 935939000 Hz atau ARFCN 5.

Gambar 21. *Email* Diterima di Gmail

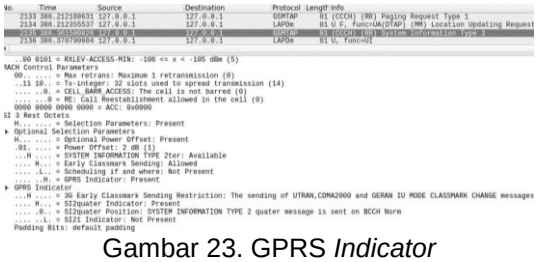
Hasil tangkapan sinyal dan *decoding* oleh RTL-SDR ditampilkan pada Wireshark. Pada *System Information Type* 3 berisi

informasi identitas *channel* atau *cell identity* dengan kode 11603. Kemudian, terdapat informasi *Location Area Identification* (LAI) dengan MCC 510 (Indonesia), MNC 10 (Telkomsel), dan LAC 12126 (Gambar 22).



Gambar 22. Informasi LAI

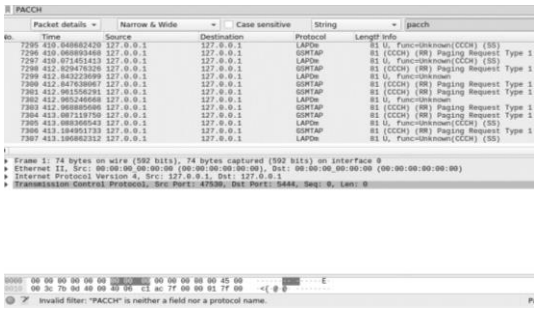
Proses *decoding* GPRS pada metode ini dilakukan dengan cara yang sama dengan proses *decoding* GSM, yaitu menggunakan Gr-GSM dari Airprobe dan Wireshark. *Broadcast Channel* GPRS menggunakan PBCCH (*Packet Broadcast Control Channel*), sedangkan GSM salah satunya menggunakan BCCH (*Broadcast Control Channel*). Namun, GPRS juga dapat menggunakan GSM BCCH sebagai PBCCH. Oleh karena itu, *System Information Type* 3 yang merupakan hasil *decoding* dengan filter BCCH mempunyai informasi GPRS *Indicator* yang menunjukkan penggunaan GPRS aktif pada BCCH seperti ditunjukkan pada Gambar 23.



Gambar 23. GPRS *Indicator*

Frekuensi *air interface* yang digunakan pada GPRS bernilai sama dengan GSM, namun mempunyai *timeslot* yang berbeda. Maka, ARFCN perangkat yang digunakan GPRS adalah 5 mengikuti ARFCN yang digunakan pada GSM. Selain itu, GPRS menggunakan layer MAC (*Medium Access Control*), RLC (*Radio Link Layer*), dan GSM-RF (*Physical Layer*) pada *air interface*, sedangkan GSM menggunakan layer RR (*Radio Resource*), protokol LAPDm (*Link Access Protocol for the ISDN D-channel*), dan GSM-RF (*Physical Layer*).

Gr-GSM dapat me-*decode* BCCH pada *Broadcast Control Channel* dan SDCCH pada *Dedicated Control Channel*. Untuk menampilkan layer MAC/RLC, dilakukan dengan filter *Packet Dedicated Control Channel*, karena MAC/RLC termasuk ke dalam *Packet Dedicated Control Channel* PACCH (*Packet Associated Control Channel*). PACCH adalah *bi-directional dedicated channel* yang membawa informasi seperti *channel assignment*, *power control*, dan *acknowledgement* penerimaan data GPRS. Tetapi, Gr-GSM tidak dapat me-*decode* PACCH, oleh karena itu Gr-GSM tidak dapat menampilkan layer MAC/RLC dan *timeslot* GPRS (Gambar 24).



Gambar 24. Tidak Terdapat *Decoding* PACCH

Untuk keamanan data pada GPRS berbeda dengan keamanan data GSM. Algoritma A5/1 hanya digunakan pada GSM untuk mengenkripsi data pada *air interface*, namun pada GPRS menggunakan algoritma GEA3 dalam enkripsi data. Oleh karena Gr-GSM tidak dapat melakukan *decoding* PACCH, maka algoritma keamanan GPRS juga tidak dapat ditampilkan pada Wireshark, sehingga proses dekripsi data GPRS tidak dapat dilakukan.

KESIMPULAN

1. Keakuratan keamanan data pada jaringan seluler masih cukup aman berdasarkan pembagian *timeslot* dan pola transmisi data pada *air interface* yang berlangsung cepat, sehingga mempersulit *sniffer* dalam menangkap dan me-*decode* sinyal. Penggunaan *GSM Frame Number* yang terus berubah pada setiap *frame* dapat membantu memperkuat keamanan data. Selain itu, penggunaan TMSI juga dapat melindungi *user* dari *sniffing*, karena informasi perangkat yang ditampilkan bukan informasi asli perangkat.
2. Penggunaan algoritma A5/1 untuk mengenkripsi sinyal GSM masih kurang aman, karena algoritma A5/1 masih

dapat didekripsi menggunakan RTL-SDR dan *tools decoding* sinyal.

3. Untuk keamanan pengiriman paket data pada GPRS masih aman diterapkan, karena *timeslot*, kanal logika, dan algoritma keamanan antara GSM dan GPRS berbeda. Sedangkan *tools decoding* dan RTL-SDR hanya dapat melakukan penangkapan dan *decoding* sinyal dengan kanal logika GSM.
4. Saran pada sistem keamanan data seluler ini sebaiknya digunakan algoritma keamanan selain A5/1 pada sistem GSM untuk menghindari serangan yang terjadi pada *air interface*.

DAFTAR PUSTAKA

Apriyanti, Y., Juhana, T., Hamidi, E.A.Z., 2016, Sniffing Sinyal GSM Dengan RTL-SDR, GNU Radio dan Wireshark, *SENTER 2016: Seminar Nasional Teknik Elektro 2016*, 26-27 November 2016, 78-85.

Laufer, C., 2014, *The Hobbyist's Guide to the RTL-SDR: Really Cheap Software Defined Radio*, RTL-SDR.com.

Ramadhan, S.A., Rizal, M.F., dan Rosmiati, M., 2018, Implementasi GNURADIO GR-DVBT2 Untuk Decoding Sinyal Televisi Digital, *e-Proceeding of Applied Science*, Volume 4, 1949-1957.

Rivaldy, B., R., Andrian, H.R., dan Rizal, M.F., 2017, Implementasi Gr-GSM Untuk Decoding Komunikasi GSM Terenkripsi, *e-Proceeding of Applied Science*, Volume 3, 1822-1832.

Wardhana, L. dan Makodian, N., 2010, *Teknologi Wireless Communication dan Wireless BroadBand*, Edisi 1, Yogyakarta: Penerbit Andi.

BIODATA PENULIS

Yulivia Rhadita Savitri, lahir di Palembang pada tanggal 22 Januari 1999. Saat ini tercatat sebagai Mahasiswa pada Program Studi Teknik Telekomunikasi, Jurusan Teknik Elektro, Politeknik Negeri Sriwijaya.

Sopian Soim, S.T., M.T., menyelesaikan pendidikan S1 dari Universitas Sriwijaya, dan pendidikan S2 dari Institut Teknologi Sepuluh Nopember Surabaya. Saat ini tercatat sebagai Dosen Tetap pada Program Studi Teknik Telekomunikasi, Jurusan Teknik Elektro Politeknik Negeri Sriwijaya.

Mohammad Fadhli, S.Pd., M.T., menyelesaikan pendidikan S1 di Universitas Negeri Padang, dan pendidikan S2 dari Institut Teknologi Sepuluh Nopember Surabaya. Saat ini tercatat sebagai Dosen Tetap pada Program Studi Teknik Telekomunikasi, Jurusan Teknik Elektro Politeknik Negeri Sriwijaya.