Jurnal Ilmiah
ILMU KOMPUTER
Universitas Udayana

Vol. IX, No. 2, September 2016 **ISSN 1979 - 5661**

RANCANG BANGUN APLIKASI *WIRELESS* PENETRATION TEST PADA SISTEM OPERASI LINUX

Aditya Dwipayana [1], I Made Oka Widyantara[2], Ni Made Ary Esta Wirastuti[3]
Jurusan Teknik Elektro dan Komputer Fakultas Teknik Universitas Udayana
Jln. Jalan Kampus Bukit Jimbaran 80361 INDONESIA
Email: aditya.dwipayana@gmail.com[1], oka_widyantara@yahoo.com[2], arydev_02@yahoo.com[3]

ABSTRAK

Algoritma enkripsi adalah pertahanan pertama terhadap serangan jaringan. *Wired Equivalent Privacy* (WEP) dan *Wi-Fi Protected Access* (WPA) digunakan untuk mengamankan jaringan pada standar 802.11, akan tetapi enkripsi ini memiliki banyak kelemahan. Aplikasi-aplikasi penetrasi berbasis linux seperti *aircrack, airodump, aireplay* mampu memanfaatkan kelemahan tersebut untuk dapat terkoneksi ke dalam jaringan dan bahkan mencuri data. Sayangnya aplikasi penetrasi berbasis *text* mengakibatkan kesulitan tersendiri bagi pengguna umum karena harus mengingat format perintah pada setiap aplikasi. Penelitian ini berhasil menyederhanakan penggunaan aplikasi penetrasi berbasis *text* dengan cara membangun sebuah antarmuka menggunakan *QTdesigner* serta memberikan *review* proses penetrasi terhadap keamanan WEP dan WPA menggunakan antar muka yang diusulkan.

Kata Kunci: *Encryption, 802.11, WEP, WPA, Aircrack, QTdesigner, Python.*

ABSTRACT

Encryption algorithm is the first defense against network attacks. Wired Equivalent Privacy (WEP) and Wi-Fi Protected Access (WPA) is used to secure 802.11-based networks, but these encryption has many weakness. Linux's tools such as aircrack, airodump, aireplay are able to exploit that weakness to gain access to the network and even steal the data. Unfortunately, text based application causing its own complexcity for common user because they must remember the format command on each application. This study managed to simplify the use of text based penetration application by build an interface using QTdesigner as well as provide a review of WEP and WPA penetration process using proposed interface system.

Keywords: *Encryption, 802.11, WEP, WPA, Aircrack, QTdesigner, Python.*

1 PENDAHULUAN

Wireless Local Area Network (WLAN) adalah suatu jaringan area *local* tanpa kabel dimana media transmisinya dapat berupa frekuensi radio (RF) dan *infrared* (IR) (Widyantara et al, 2010). Tingkat ancaman pada teknologi *wireless* semakin berkembang sejalan dengan perkembangan teknologi *wireless*. Sifat jaringan *wireless* yang *mobile* memicu meningkatnya potensi ancaman keamanan yang lebih besar dibandingkan jaringan kabel. Sehingga untuk mencegah akses

yang tidak sah, IEEE memperkenalkan standar keamanan nirkabel pertama pada tahun 1999 yang disebut *Wired Equivalent Privacy* (WEP). Algoritma enkripsi WEP ini menggunakan algoritma *Rivest Chiper* 4 (RC4) dari RSA Data Security.Namun, algoritma enkripsi ini tidak bertahan lama.Pada tahun 2001 ditemukannya celah keamanan pada algoritma *key scheduling* RC4 (Fluhrer et al, 2001). Sejak saat itu standar WEP dikenal dengan enkripsi *protocol* yang lemah. Sehingga pada tahun 2003, Standar ini kemudian digantikan oleh *Wi-Fi Protected Access* (WPA) dan berlanjut digantikan oleh standar 802.11i atau yang lebih dikenal dengan nama WPA2 pada tahun 2004.Algoritma WPA ternyata juga memiliki kelemahan pada Authentifikasi *Pre-shared key* dan *Temporal Key Integrity Protocol*.

Aircrack, *airodump*, *aireplay* adalah beberapa aplikasi yang berfungsi untuk melakukan penetrasi pada jaringan *wireless*. Aplikasi ini mencoba mengekploitasi kelemahan-kelemahan yang hadir pada alogitma enkripsi WEP dan WPA. Namun sayangnya, aplikasi tersebut dijalankan melalui *commandline* sehingga pengguna harus mengetahui baris perintah yang harus dijalankan. Penelitian ini mengusulkan sebuah *interface* untuk mempermudah pengguna dalam melakukan penetrasi terhadap suatu jaringan *wireless*. *Interface* pada penelitian ini dibangun dengan memanfaatkan aplikasi *Qt-designer* yang menghubungkan *interface* dengan aplikasi berbasis *text*.

2 KELEMAHAN PADA STANDAR 802.11

De-authentication adalah contoh sebuah serangan yang dapat bekerja pada *protocol* WEP bahkan WPA (Tews et al, 2009). Serangan *De-authentication* ini bekerja dengan cara mengirim banyak paket *De-authentication* ke jaringan *wireless* sehingga mengacaukan *wireless service client*. Serangan ini dapat mengakibatkan terputusnya koneksi seluruh pengguna *wireless* yang lain :

2.1 Kelemahan Algoritma Enkripsi WEP

WEP adalah algoritma enkripsi yang dikembangkan oleh IEEE. Algoritma enkripsi WEP menggunakan algoritma RC4 dan menggunakan 2 buah ukuran kunci yakni 40 bit dan 104 bit. Pengguna *wireless* yang dapat terkoneksi dengan *accesspoint* adalah hanya pengguna yang mempunyai kunci rahasia yang sama. Metode enkripsi pada algoritma WEP disebut simetrik, karena WEP menggunakan kunci yang sama dalam mengenkripsi dan mendekripsi data.

Algoritma enkripsi WEP memiliki berbagai kelemahan antara lain :

- Algoritma RC4 yang digunakan dalam WEP dapat dipecahkan.
- WEP masih menggunakan kunci yang bersifat statis.

- Kelemahan pada *initializationvector* (IV).
- Kelemahan pada integritas pesan *Cyclic Redundancy Check* (CRC-32).

Serangan-serangan terhadap kelemahan WEP antara lain :

- FMS*Attack*

  Serangan ini disebut FMS *attack* karena diambil dari singkatan nama ketiga penemu celah keamanan ini yakni Fluhrer, Mantin, dan Shamir. FMS *attack* dilakukan dengan cara mengumpulkan *initialization vector* lemah dalam jumlah banyak. Semakin banyak IV lemah yang didapat maka semakin cepat *cracking* dapat dilakukan.
- Choping *Attack*

  Pertama kali ditemukan oleh seseorang dengan identitas di internet bernama h1kari. Tidak seperti FMS *attack*, teknik ini hanya membutuhkan IV unik yang kemudian dgunakan pada proses *cracking* kunci WEP. IV unik mampu mengurangi kebutuhan IV lemah dalam melakukan *cracking* WEP.

Kedua serangan diatas membutuhkan waktu dan paket data yang cukup, sehingga untuk mempercepat proses *cracking*, para *cracker* biasanya melakukan

2.2 Kelemahan Algoritma Enkripsi WPA

Wi-Fi Protected Access (WPA) adalah sebuah algoritma enkripsi yang dikembangkan oleh *Wi-Fi Alliance* sebagai upaya penyempurnaan kelemahan-kelemahan yang ditemukan pada algoritma enkripsi *Wired Equivalent Privacy* (WEP). Perubahan yang paling signifikan adalah penambahan *Pre-Shared Key* (PSK) dan *Temporal Key Integrity Protocol* (TKIP). Penambahan TKIP tersebut ternyata tidak menjamin algoritma enkripsi WPA dari kelemahan-kelemahan.

Kelemahan-kelemahan pada algoritma enkripsi WPA antara lain :

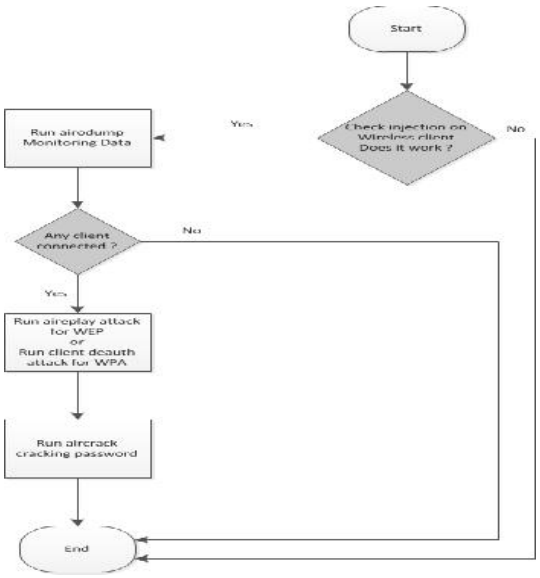
- Authentifikasi *Pre-shared key* (PSK) yang digunakan pada algoritma enkripsi WPA sangat rentan terhadap serangan *offline dictionary attacks*.
- *Temporal Key IntegrityProtocol* (TKIP) yang digunakan pada algoritma enkripsi WPA juga rentan terhadap serangan *ChopChop attack* karena sama-sama menggunakan RC4 seperti WEP.

3 SKEMA SISTEM

Perancangan aplikasi *wireless penetration test* dibangun pada sistem operasi *linux*. Sistem operasi yang digunakan adalah *Linux Ubuntu* 8.04. Perancangan sistem ini dilakukan melalui 4 tahap yakni pembuatan *flowchart* sistem, pembuatan *Diagram context*, pembuatan DFD *level* 0 dan pembuatan DFD *level* 1. Keempat proses tersebut mampu mewakili langkah-langkah penetrasi algoritma enkripsi WEP dan WPA.

3.1 *Flowchart Sistem*

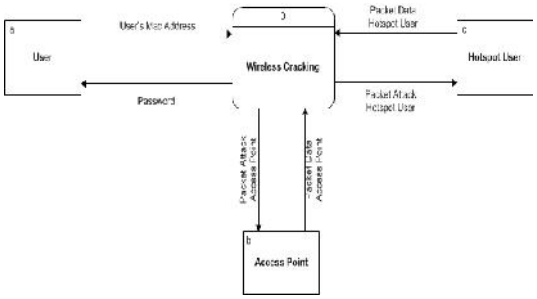
Flowchart berfungsi untuk menggambarkan alur data secara keseluruhan pada suatu sistem. *Flowchart* dari sistem *wireless penetration test* ditunjukan pada Gambar 1.



Gambar 1. *Flowchart* Sistem.

3.2 *Diagram* Con*text*

Diagram context untuk sistem *wireless penetration test* ditunjukkan pada Gambar 2.

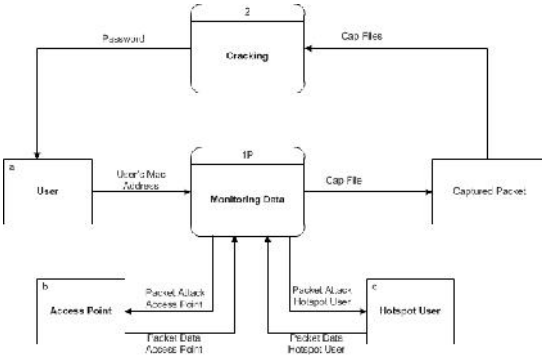


Gambar 2. *Diagram Context*

Diagram context adalah *Diagram level* tertinggi dari *Data Flow Diagram* (DFD). *Diagram context* menggambarkan hubungan sistem dengan entitas-entitas eksternal serta menunjukan masukan/*input* dan keluaran/*output* sistem secara keseluruhan.

3.3 DFD *Level* 0

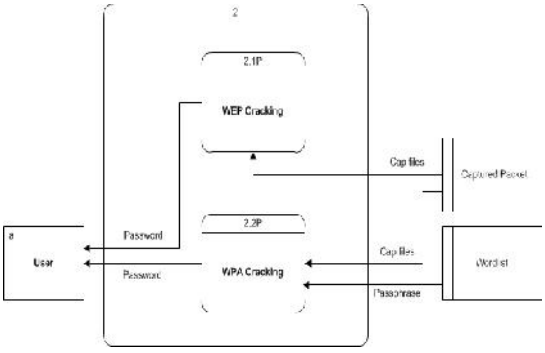
DFD *level* 0 berfungsi untuk menggambarkan *internal* sistem secara keseluruhan. Proses *monitoring* data adalah proses untuk me*monitoring*/meng*capture* paket data yang dikirim oleh *accesspoint*. Sedangkan proses *cracking* adalah proses pencarian *password* terhadap data yang telah dikumpulkan.



Gambar 3. DFD*Level* 0

3.4 DFD *Level* 1

Pada DFD *level* 1, proses *cracking* dijabarkan menjadi 2 proses yang lebih kecil, yakni proses WEP *cracking* dan WPA *cracking*.



Gambar 4. DFD *Level* 1.

Pada proses WEP *cracking* (2.1), masukan/*input* dari proses ini adalah *capfile* dari hasil proses *monitoring* data. Sedangkan Keluaran/*output* dari proses ini adalah *password* hasil *cracking*. Untuk proses WPA *cracking* (2.2) terdapat 2 masukan/*input* untuk proses ini yakni, *wordlist* dan *capfile*. Keluaran/*output* dari proses ini adalah *password* hasil *cracking*.

4 IMPLEMENTASI SISTEM

Secara garis besar implementasi sistem *wireless penetration test* ini terbagi kedalam 2 proses inti yaitu proses *monitoring* paket data dan proses *cracking*. Seluruh proses tersebut ditampilkan dalam bentuk antarmuka grafis. Langkah-langkah untuk melakukan *penetrasi wireless cracking* dibangun berurutan serta dipisahkan ke dalam lima *menu* berbeda yang terdiri dari *Setup, Scan*, WEP *cracking*, WPA *Cracking* dan *Cracking Password* sehingga memudahkan pengguna dalam menggunakan aplikasi.

Langkah pertama dalam melakukan proses penetrasi terhadap suatu jaringan *wireless* adalah melakukan konfigurasi sistem pada *menu setup* seperti ditunjukan pada Gambar 5. Pada *menu* ini, pengguna dapat mengkonfigurasi :

- **Lokasi *file***

  Lokasi *file* adalah lokasi tempat menyimpan *file log* sistem, *file backup mac address*, *file captured packet*, dll. Tombol *delete old* data berfungsi untuk menghapus seluruh *log* terakhir dengan cara menjalankan perintah *rm –f *.cap *.csv *.xor *.netxml.*

- **Konfigurasi *wireless adapter***

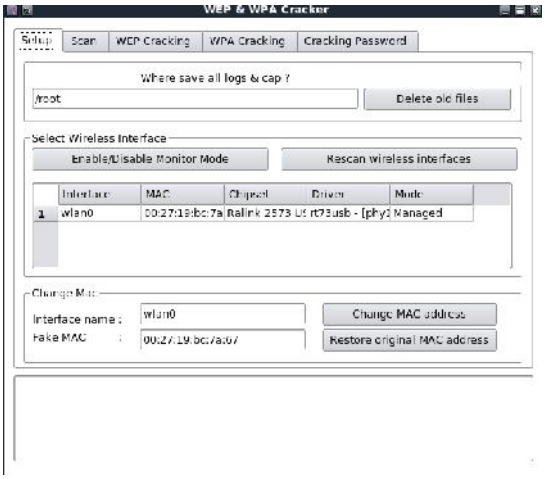
  Pengguna dapat memilih *wireless adapter* yang akan digunakan serta mengaktifkan/ mematikan *mode monitor* pada *wireless adapter* tersebut dengan bantuan aplikasi *airmon* dengan menjalan perintah *airmon-ng start wlan0.*

- **Mengganti *Mac address***

  Pengguna dapat mengganti *mac address* dari *wireless* card yang digunakan untuk menjaga privasi (menyembunyikan jati diri) pengguna tersebut. Perintah yang dijalankan oleh sistem adalah *macchanger –mac mon0.*

- **Restore *Mac address***

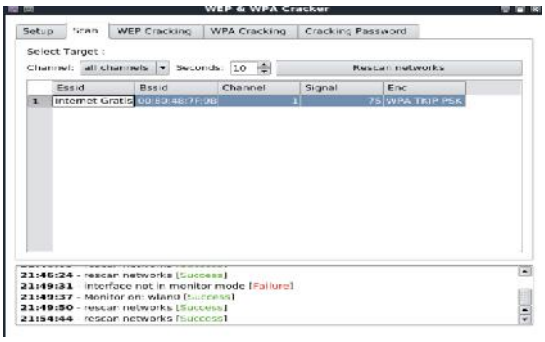
  Seperti namanya, tombol ini berfungsi untuk mengembalikan *mac address* dari *wireless* card ke keadaan semula.



Gambar 5. Tampilan *menusetup*

Gambar 6 menampilkan *men uscan*. *Menu scan* berfungsi untuk melakukan *scaning* terhadap jaringan *wireless* yang

tersedia. Pada *menu scan*, pengguna dapat memilih *channel wireless* serta *delay time* yang akan digunakan. Proses *scaning* dilakukan melalui tombol *rescan network* yang kemudian menjalankan perintah *airodump-ng –output-format csv –write '/tmp/wepwpacracker'*.



Gambar 6. Tampilan *menuscan*.

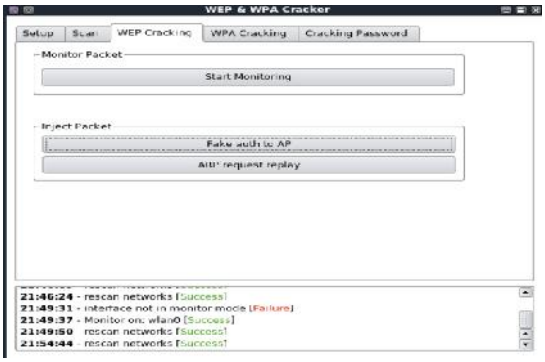
Gambar 7 menunjukan tampilan *menu* WEP *cracking*. *Menu* ini berfungsi untuk melakukan *cracking* terhadap jaringan *wireless* yang menggunakan tipe algoritma enkripsi WEP. *Menu* ini dibagi menjadi 2 bagian penting yaitu *monitoring packet* dan injeksi *packet*

Pada bagian *monitorin gpacket* terdapat tombol *start monitoring* yang berfungsi untuk meng*capture* data yang dikirim oleh accesss *point* dengan cara menjalankan perintah *airodump-ng –c Channel –w File –bssid MacAP Wireless Adapter*. Perintah tersebut akan memunculkan *pop-up shell* seperti ditunjukan pada Gambar 8.

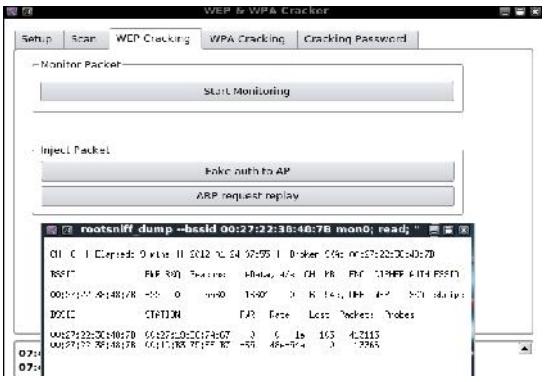
Pada bagian *inject packet* terdapat 2 tombol, yakni *Fake auth to AP* dan *ARP request replay*. Tombol *fake auth to AP* berfungsi untuk melakukan authentifikasi palsu ke *access point* dengan cara menjalankan perintah *aireplay -1 0 –eSSIDAP –a MacAP –h MacWirelessAdapterWirelessAdapter*.

Sedangkan tombol *ARP request replay* berfungsi untuk melakukan serangan *arp request* terhadap accesss *point* dengan tujuan mempercepat proses pengumpulan *Initialization Vector* (IV). Tombol ini akan menjalankan perintah *aireplay -3 –b MacAP –h WirelessAdapterMacWirelessAdapter*.

Perintah tersebut memunculkan *pop-up shell* seperti yang ditunjukan pada Gambar 9.



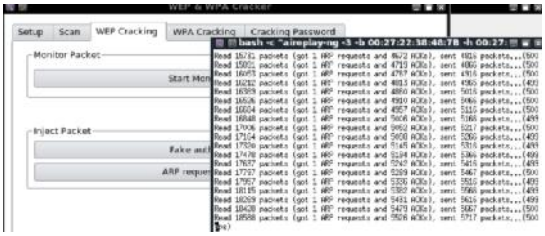
Gambar 7. Tampilan *form* WEP *Cracking*



Gambar 8. Tampilan *Start Monitoring*.

Setelah menjalankan *script* untuk me*monitoring packet*, selanjutkan

pengguna dapat melakukan serangan dengan meng*klik* tombol *Fake Auth to AP* yang diikuti dengan meng*klik* tombol *ARP Request Replay*. Maka akan muncul sebuah *pop-up shell* seperti gambar di bawah ini.

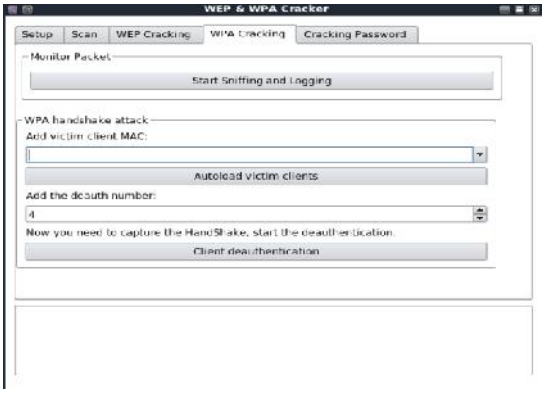


Gambar 9. Tampilan *ARP request replay*.

Menu WPA *cracking* seperti ditunjukan pada Gambar 10 digunakan untuk melakukan *cracking* terhadap jaringan *wireless* yang menggunakan algoritma enkripsi WPA. *Menu* ini dibagi menjadi 2 bagian penting, yakni bagian *monitoring packet* dan *wpa handshake attack*. Pada bagian *monitoring packet* terdapat tombol *start sniffing and logging* yang berfungsi untuk meng*capture* data yang dikirimkan oleh *access point* dengan cara menjalankan perintah *airodump-ng –c Channel –w File –bssid MacAP WirelessAdapter*.

Pada bagian WPA *handshake attack* terdapat 2 tombol yakni tombol *autoload victim clients* yang berfungsi untuk memutus koneksi *wireless* penguna lain dengan perintah *aireplay -0 AttackNumber –a MacAP –c MacClient WirelessAdapter*. Serangan ini dimaksudkan agar *client* yang terputus berusaha terkoneksi kembali ke *access point* sehingga sistem dapat meng*capture* paket *handshake* yang terjadi.

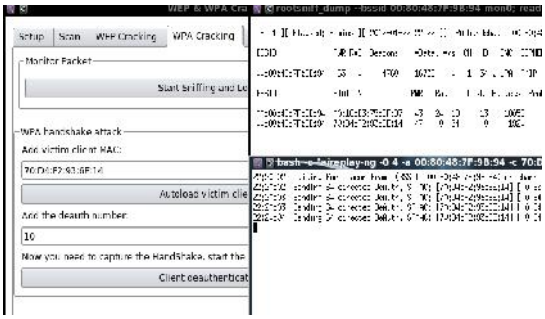
Gambar 11 menunjukan tampilan *pop-upshell* ketika pengguna meng*klik* tombol *start sniffing and logging*.



Gambar 10. Tampilan *menu* WPA *Cracking*.

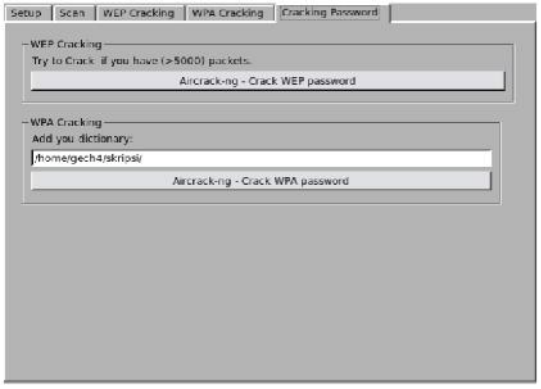
Gambar 11. Tampilan *pop-upshell start* sniffing *and logging.*

Gambar 12 menunjukan tampilan ketika pengguna meng*klik* tombol *client deauthentication* yang menyebabkan *client* tersebut terputus dan melakukan authentifikasi ulang ke accesss *point*. Ketika sistem mendeteksi adanya paket *handshake* maka akan terlihat pada pojok kanan atas pada bagian WPA *handshake*.



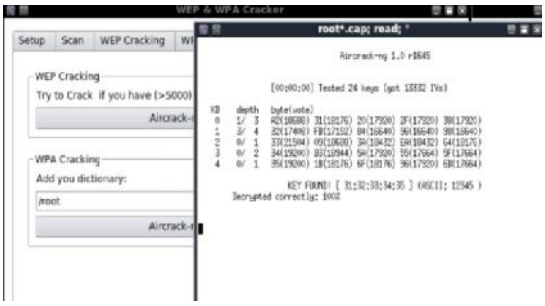
Gambar 12. Tampilan *start* sniffing and logging.

Menu terakhir adalah *menu cracking password*. Tampilan *menu* ini ditunjukan pada Gambar 13.*Menu* ini berfungsi untuk melakukan *cracking password* terhadap data yang telah dikumpulkan sebelumnya. Pada *menu* ini terdapat 2 tombol untuk melakukan *crackingpassword* sesuai dengan tipe algoritma enkripsinya.



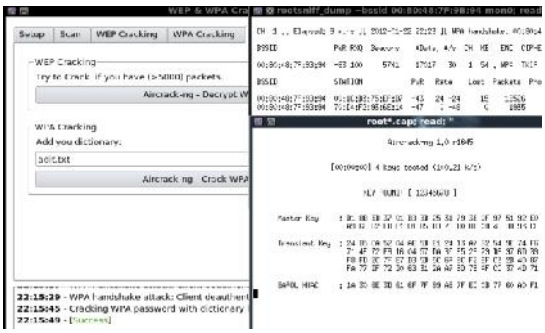
Gambar 13. Tampilan *menu Cracking Password*.

Cracking terhadap WEP dapat dilakukan setelah *packet* data yang terkumpul menunjukan lebih dari 5000.Dengan menekan tombol *Aircrack-ng – Crack WEP password*. Tombol ini akan menjalankan perintah *aircrack –z –b MacAP *.cap*.



Gambar 14. Tampilan WEP *cracking*.

Gambar 14 adalah tampilaan sistem ketika berhasil melakukan *cracking* terhadap WEP ditandai dengan tulisan *KEY FOUND*. Berbeda dengan *cracking* WEP. *Cracking* WPA dilakukan dengan cara menyediakan *wordlist* yang akan digunakan dalam melakukan WPA *cracking* dan kemudian menekan tombol *Aircrack-ng – Crack* WPA *password* seperti ditunjukan pada Gambar 15.



Gambar 15. Tampilan WPA *cracking*.

5 SIMPULAN

Paper ini telah memaparkan tahapan-tahapan perancangan *interface* aplikasi *wirelesspenetration test* dengan cara memanfaatkan *tab widget* yang teratur untuk memisahkan langkah-langkah penggunaan aplikasi. Sedangkan integrasi antara aplikasi berbasis *text* dengan *interface* dilakukan dengan cara

membangun sebuah "*slot*" pada setiap tombol. Setiap tombol yang ditekan akan memerintahkan sistem operasi untuk menjalankan perintah/*commandtextbased*. *Interface* yang dihasilkan mampu mempermudah proses penetrasi pada jaringan *wireless* dengan tetap mengacu pada langkah-langkah yang sama pada aplikasi berbasis *text* tanpa harus menghafal baris perintah seperti pada *mode text*.

6 DAFTAR PUSTAKA

A. Bittau, M. Handley, J. Lackey, "The final nail in WEP's coffin," in IEEESymposium on Security and Privacy, 2006.

Fluhrer, S., Mantin, I., & Shamir, A., "Weaknesses in the key scheduling algorithm of RC4," in International Workshop on Selected Areas in Cryptography. 2001. pages 1-24.

G. Mouhcine, L. Aboubakr and B. Amine Benamrane, "Wireless Networks Security: Proof of ChopChop Attack," in World of Wireless, Mobile and Multimedia Networks, 2008.

J. Bellardo and S. Savage, "802.11 Denial-of-Service Attacks: Real Vulnerabilities and Practical Solutions," in USENIX Security Symposium,2003, pages 15-28.

P. S. Ambavkar, P. U. Patil, B. B. Meshram, and P. K. Swamy, " WPA exploitation in the world of wireless network," *Int J Adv Res Comput Eng Technol*. 2012.

Tews, E., & Beck, M., "Practical attacks against WEP and WPA", in Proceedings of the second ACM conference on Wireless network security. 2009. pages 79-86.

Widyantara, I. M. O., Cahyono, B. D., Setiawan W., "Analisa Horizontal Handover Terhadap QoS Layanan Streaming Multimedia E-Learning Pada Jaringan WLAN 802.11," *Jurnal Teknologi Elektro, Vol.14.* 2015.

Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE Std. 802.11, 1997.

Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Amandment 6 Medium Access Control (MAC) Security Enhancements, IEEE Std. 802.11i, 2004.

S'to, *Wireless* Kung fu : Networking & Hacking, Jasakom, 2007.

(2009) Aircrack website. [Online]. Available: https://www.aircrack-ng.org/doku.php?id=korek_chopchop.

(2012) Qt Designer Manual. [Online]. Available: http://doc.qt.io/qt-4.8/designer-manual.html.