PERBANDINGAN KINERJA WIRESHARK DAN CAIN DALAM ANALISIS JARINGAN INTERNET

Edy Victor Haryanto¹, Anderian²

^{1,2} STMIK Potensi Utama Jl. K. L Yos Sudarso Km.6,5 No.3A Tanjung Mulia Medan Email : edyvictor@gmail.com, the.dark.heart19@gmail.com

Abstrak

Pengamanan data pada saat sangat dibutuhkan sekali, apalagi dalam hal jaringan komputer, siapapun dan dimanapun jaringan kita dapat dibobol atau diakses orang lain tanpa kita sendiri ketahui, maka dari itu perlu diketahui bagaimana seseorang dapat mengakses jaringan komputer kita atau orang lain, dalam penelitian ini akan dibahas mengenai metode wireshark dan cain dalam penggunaan jaringan internet

Kata Kunci : Jaringan komputer, sniffing, jaringan nirkabel, Cain.

1. PENDAHULUAN

Seiring dengan semakin berkembangnya teknologi *internet*, kejahatan yang memanfaatkan teknologi ini juga semakin meningkat. Hal ini ditambah lagi dengan semakin banyaknya peredaran aplikasi gratis yang dapat digunakan untuk melancarkan usaha pembobolan suatu sistem berbasis teknologi jaringan internet. Akan tetapi *internet* bukan tanpa resiko, karena maraknya kegiatan *cyber crime* akhir-akhir ini yang bisa mencuri data dan penyadapan transmisi pada jaringan. Oleh karena itulah dibutuhkan suatu aplikasi penganalisa jaringan yang bernama Cain dan Wireshark.

Cain adalah suatu aplikasi pemulihan sandi untuk sistem operasi Microsoft. Untuk itu Cain dapat melakukan pemulihan password dengan mengendus protocol jaringan yang ada, membuka password yang didekripsi melalui kamus, merekam percakapan VoIP, memecahkan kode secara acak, dan memulihkan kunci jaringan nirkabel.

2. Landasan Teori

2.2 Aplikasi Wireless LAN 2.2.1 Akses Role

Wireless LAN kebanyakan menyebar dalam suatu lapisan akses,maksudnya mereka digunakan sebagai suatu titik masukan ke dalam suatu kabeljaringan. Di masa lalu, akses telah digambarkan sebagai dial-up, ADSI, kabel / telegram, selular, Ethernet, Token Ring, Frame Relay, ATM, dan lain lain. Wireless cara sederhana yang lain untuk para user dalam mengakses jaringantersebut. Wireless LAN adalah lapisan jaringan Data-Link seperti cara akses semuanya hanya mendaftar saja. Dalam kaitan dengan kecepatan, jaringan nirkabel tidaklah tepat diterapkan dalam distributor atau sebagai inti dalam jaringan.



Gambar 1. Akses role dari wireless LAN

3. Analisis Hasil Pengujian

3.1 Hasil Pengujian Cain

Pengujian yang dilakukan disini adalah pada Wifi yang ada, yaitu SSID *Wi-Fi*. Ujicoba yang dilakukan adalah seperti berikut :

Untuk melakukan analisis dengan Cain, maka terlebih dahulu dibuka program aplikasi Cain, klik kanan pada *shortcut* Cain, lalu pilih *Run As Administrator*.



Gambar 7. Membuka aplikasi Cain pada Windows 7

Hasilnya dapat dilihat seperti pada gambar berikut



Gambar 8. Tampilan Menu Utama Cain

Selanjutnya, pilih menu *Configure*, lalu akan muncul menu *Configure* seperti berikut :

Decoders 9 Network 19	Sniffer 🖉 Cracker 😋 Traceroute 🛄 CCDU 😗 Wireless 🚯 Query	
Carlad Paramonia Via Screen Via S	Press the + builden on the bucher to during the Protoched Norsege	

Gambar 9. Tampilan Letak Menu Configure pada Cain

Penulis akan melakukan suatu teknik penyadapan yang dinamakan *ARP Poisoning* atau yang lebih dikenal dengan istilah *Man In The Middle Attack*. Selanjutnya klik *configure* lalu tentukan interface yang digunakan, disini penulis menggunakan wireless. Untuk melihat lebih jelas apakah tersebut adalah LAN atau wireless, dapat di *scroll* ke kanan, lalu klik OK.

figuration Dialog		
Challenge Spoofing	Filters and ports	HTTP Fields
Traceroute Certificat	e Spoofing Cer	tificates Collecto
Sniffer APR (Arp Poi	ison Routing)	APR-SSL Option
Adapter	IP address	Subnet Mask
Device\NPF_{62C894F	192.168.7.7	255.255.255.
Device\NPF_{500C777	192.168.180.167	255.255.255.
-Winpcap Version 4.1.0.1753		
-Current Network Adapter -		
\Device\NPF_{500C7778-	579D-4155-869E-EEA	5A4C54C85}
WARNING !!! Only	y ethernet adapters su	oported
Options		
Start Sniffer on startup	Don't use Promis	scuous mode
Start APR on startup		

Gambar 10. Tampilan Pemilihan Interface yang Ingin Digunakan

Lalu pilih tab sniffer, dan juga klik button sniffer



Gambar 11. Menghidupkan Sistem Sniffer Pada Cain

Lalu klik tombol *add to list* yang berwarna biru untuk menambahkan host yang akan di sniffing, lalu klik ok.



Gambar 12. Memasukkan Host yang Ada Pada Jaringan

Maka Cain akan menampilkan semua host yang terdapat pada jaringan, seperti berikut :



Gambar 13. Tampilan dari Semua Host Dalam Kondisi *Idle*

Bila dilihat pada gambar di atas, maka status dari semua host adalah *idle* artinya belum ada aktifitas yang didapat oleh Cain terhadap setiap host yang ada. Untuk itu, selanjutnya blok semua host dan klik button arp yang berwarna kuning sebelah atas, sehingga status semua host menjadi *poisoning*.



Gambar 14. Melakukan ARP Poisoning pada Host yang Sudah Didapat Oleh Cain

Pilih tab password yang berada di sebelah bawah, sehingga dapat terlihat apakah ada host yang melakukan login ke dalam suatu situs. Pada gambar di bawah di dapatkan username dan password dari salah satu host yang melakukan login ke situs <u>http://www.ziddu.com</u>, dengan username <u>the.dark.heart19@gmail.com</u> dan password krm236



Gambar 15. Hasil Dari Host yang Melakukan Login

Dari hasil percobaan di atas, maka dapat diketahui bahwa Cain sangat berguna jika digunakan dengan baik oleh seorang administrator jaringan. hitung untuk kenaikan rata kuat sinyal wireless dari perangkat access point yang kuat sinyalnya dikuatkan dengan penguat sinyal vang menggunkaan kaleng minuman bekas dapat mempengaruhi penguatan sinyal sebesar 15.84158 % dari nilai kuat sinyal access point standart. Nilai sebesar 15.84158 merupakan nilai hasil rata - rata penguatan sinyal yang dilakukan terhadap 20 kali pengujian kuat sinyal. Hasil nilai prosentasi 15.84158 % dapat disimpulkan bahwa kaleng minuman bekas dapat menaikkan kuat sinyal transmisi wireless sebesar prosentasi penghitungan.

3.2. Hasil Pengujian Wireshark3.2.1. Sniffing Paket HTTP

Uji coba analisis jaringan komputer melalui komputer user/sniffer mail server dilakukan melalui sistem operasi Ubuntu Dekstop 12.04, sedangkan client menggunakan sistem operasi windows, bisa juga digunakan sistem operasi lain, tetapi penulis pada uji coba perancangan client menggunakan sistem operasi windows. Disini penulis akan menjelaskan cara melakukan *sniffing* pada protocol HTTP (*Hypertext Transfer Protocol*), pada akses point WIFI. Protokol HTTP adalah protokol yang digunakan untuk transfer data di dalam internet antara *server* dan *client*.

Langkah pertama yang harus dilakukan adalah masuk sebagai root user dari terminal linux, ketik *ctrl+alt+t* dari keyboard untuk membuka terminal, lalu ketikkan *sudo su* untuk masuk sebagai *root user*, lalu masukkan passwordnya, dan ketik *sudo wireshark* untuk memanggil aplikasi Wireshark seperti pada gambar berikut ini :



Gambar 16. Tampilan Perintah dan Menu Utama Wireshark

Setelah tampilan Wireshark muncul, pilih tab capture lalu capture option untuk memilih *interface* jaringan yang kita gunakan, disini penulis memakai LAN, sehingga *interface* dipilih adalah *eth0*.



Gambar 17. Tampilan Capture Options Wireshark

Setelah itu, maka akan muncul hasil record yang didapat secara *real time* oleh Wireshark, tetapi karena Wireshark merekam semua lalu lintas yang terdapat dalam jaringan, maka semua informasi tersebut akan langsung di *capture* dalam tampilan Wireshark. Untuk itu diperlukan *filter* untuk menentukan paket apa saja yang ingin diketahui oleh *user*. Ketikkan *http* pada kolom *filter* sehingga paket yang muncul adalah yang memakai protokol http.

coback	(Wkeshark 1.6.7)		No. 1 August and a	🚾 🛤 🗢 4 1531M 1 Andry 🗘
1000	File Edit View Go Capture Analyze Sta			
0	製廠廠的影 1 2 2 2 4 0	: 🚊 Q 🗧 🦻 🎙	F 🛓 🗐 🖪 O O O 🗹 🗃 🕅 📑 1	< 0
1	Filter: http	Expression	Clear Apply	
	No. Time Source	Destination Proto	cal Length Info	
1	53802 997.328766 192.168.182.248	uploads.riddu.com HTTP	648 GET /inages/flags.gif HTTP/1.1	
3	52805 997.329894 192.168.180.248	uploads.ziddu.com HTTP	953 GET /images/headcorner.glf HTTP/1.1	
	52810 097.333469 102.168.185.248	uploads.ziddu.com HTTP	952 GET /images/log-ziddw.gif HTTP/1.1	
20	52815 997,442734 91,189,95,83	192.168.180.248 HTTP	1454 Continuation or non-HTTP traffic	
-	52817 997.465354 01.189.95.83	192.168.180.248 HTTP	1454 Continuation or non-HTTP traffic	
	52821 997.487866 91.189.95.83	192.168.180.248 HTTP	1454 Continuation or non-HTTP traffic	
	52823 997.510572 91.185.95.83	192.168.188.248 HTTP	1454 Continuation or non-HTTP traffic	
	52825 997.556782 91.189.95.83	192.168.185.248 HTTP	1454 Continuation or non-HTTP traffic	
_	52827 997.579468 91.189.95.83	192.168.180.248 HTTP	1454 Continuation or non-HTTP traffic	
100	\$2833 997.653838 uploads.21ddu.com	192,168.180.248 HTTP	305 HTTP/1.1 200 OK (text/html)	
	52848 997.715184 192.168.185.248	uploads.ridds.com HTTP	919 GET /im_file_upload.js HTTP/1.1	
-	52847 997.716375 192.168.180.248	ugloads.ziddu.com HTTP	548 GET /images/avtor.gif HTTP/1.1	
200	52850 997.717033 192.168.180.248	usloads.ziddu.con HTTP	053 GET /inages/upload-but.gif HTTP/1.1	
<u> </u>	Frame 52833: 305 bytes on wire (2440 b)	its), 305 bytes captured (244	0 bits)	
-	Ethernet II, Src: 192.168.180.1 (80:0c	42:95:7c:5f), Dst: 192.168.1	80.248 (74:21:68:18:36:60)	
9	Internet Protocol Version 4, Src: uplo	ads.zidds.com (84.45.63.40),	Dit: 192.168.180.248 (192.168.180.248)	
	Transmission Control Protocol, Src Port	t: http (80), Dat Port: 44448	(44448), Seq: 8435, Ack: 978, Len: 239	
	[11 Reassembled TCP Segnects (8673 byte	<pre>#52741(1388), #52750(138</pre>	B), #52756(73), #52776(1388), #52780(11), #52782(1388)), #52784(11), #52785(1388), #52789(11),
	* Hypertext Transfer Protocol			
	Line-based text data: text/html			
-				
188				
100				
	0000 74 2f 68 f8 3d b0 00 0c 42 95 7c	5f 08 00 45 00 t/h.= H.	E.	
-	0010 01 23 bf e2 40 00 f8 05 b8 fb 54	2d 3f 28 c0 a8 .z@	1-2(
	France Phil Index1 - Base ambled W/D HEY2 has	ed at 22 an en an antitute //	ASTO human	
	Linkie Dwindrest Inspectionen (C). (2012 phi	vil - ancentrated expeription (rear of the part o	
	File: "/media/Heart/andri/cobaok Packet		0 Load time: 0:02.720	Profile: Default

Gambar 18. Tampilan Wireshark dengan Filter Http

Lalu setelah ditemukan paket protokol yang diinginkan, seperti pada gambar, berasal dari situs http://www.ziddu.com. klik kanan pada paket tersebut, lalu klik *Follow TCP Stream* untuk melihat detail informasi yang didapatkan dari paket tersebut. Gambar berikut merupakan isi detail paket tersebut.



Gambar 19. Tampilan Detail Paket HTTP yang Dicapture

3.2.1.1. Analisis Paket HTTP

Berdasarkan paket yang didapat, host tujuan digunakan yang adalah http://www.ziddu.com, hal tersebut berdasarkan pada baris informasi paket yaitu Referer: http://www.ziddu.com/upload.php. Tanggal pengambilan paket terlihat pada baris Date: Thu, 07 Nov 2013 16:16:36 GMT, yaitu hari Kamis, 7 November 2013. Server host ziddu menggunakan Apache, sesuai dengan baris informasi Server: Apache. Pada baris yang diseleksi, terdapat informasi mengenai akun pengguna yaitu Poo22. dan email username • the.dark.heart19@gmail.com. Hal tersebut merupakan salah satu kelemahan protokol HTTP, karena tidak memiliki enkripsi, sehingga seringkali username dan password pengguna dapat tertangkap oleh pengguna yang tidak berhak. Untuk itu tidak ada, transaksi internet banking, dan pembayaran lainnya menggunakan protokol ini.

3.3. Sniffing Paket HTTPS (HTTP Secure)

Untuk mengetahui perbedaannya dengan protokol HTTP, maka penulis akan menjabarkan

analisis paket pada protokol HTTPS. Masih dalam *live capture* yang sama, penulis akan menganalisis salah satu paket yang terekam dari situs <u>http://www.facebook.com</u> yaitu paket *logout* dari salah satu user, seperti terlihat dalam gambar IV.5 berikut ini.

obask	[Wireshark 1.6.7]			🖾 🛤 🕴 🖓 📢 9:31 PM 🗜 Andry
9	₩ ₩ ₩ % % <mark>`````</mark> X C ₩	Q 🔶 🤿 🎙 🛔	. 📑 🖬 🛛 🖬 🖉 🖉 🖉	<u>™</u> × ⊚
	Filter: tcp.stream eq 530	• Expression Clea	Apply	
	No. Time Source Destin	ition Protocol L	endth info	
	1930 344, 346,347 192, 346, 146, 346 192, 346, 146, 346 1935 347, 335, 335, 336, 346, 346, 346, 346, 346, 346, 346	House Process Process House Process Process House Process HouseProcess HouseProcess HouseProcess HouseProcese	14 1017 - 1419 (311) Seep S March March 1480 L 178 1419 - 31170 (3170, KGS) Seep C Asi'l Asi 168 047 - 1410 (316) Seep C Asi'l Asi'l March 164 047 - 14102 (316) Seep C Asi'l Asi'l March 14170 - 1412 (316) Seep C Asi'l Asi'l Asi'l Asi'l Asi'l Asi'l 1410 (316) Seep C Asi'l	entro 885-1440 500, 4888-1 Sval248854 TS kikelijito unem 805-1440 Svit Tsal248854 TS dia Levo Tsal24855 Star248557122 1.1 kikelijito unem 805-1456 Svit Tsal24855345 Svitekliko unem 805-1456 Svit Tsal24855345 Svitekliko unem 805-1456 Svit Tsal24855345 Svitekliko unem 805-1456 Svit Tsal24855345 Svitekliko unem Tsal24853455 Sviteklikova 40000 and 1440 Sviteklikova 1500-1500-1500-1500-1500 sviteklikova 1500-1600 Sviteklikova 40000-4000
	20068 369.060520 star.c10r.facebook.cor192.16 > Internet Protocol Version 4, Src: 192.168.180.	8.180.248 TCP 248 (192.168.180.248), Ds1	66 http > 36170 [ACK] Seg=411 Ack=1604 : star.c10r.facebook.com (31.13.68.8)	Win=5803 Len=0 TSval=865578923 TSecr=29045
	Transmission Control Protocol, Src Port: 36170	(36170), Dst Port: http (80), Seq: 739, Ack: 411, Len: 865	
	* Hypertext Transfer Protocol			
8	POST /Logeut.php #TTP/1.1\r\n			
	<pre>mess: www.lacebook.com/vm User-Agent Morslla/30 (011; Ubuntu; Linux i Accept: text/html,application/attml+xml,appli Accept-Linguage: en-us_mcige0.5V/in Accept-Longuage: pzi,b_dellate/vm Connection; keep-alive/v/m</pre>	686; rv:11.0) Gecka/20100 cation/wnl;q=0.9,*/*;q=0.1	01 Firefox/11.0\r\n Ur\n	
1	Referer: http://www.facebook.com/\r\n			
1	[truncated] coskie: datrikuolugidad-knipparta	1/W2; 1/POQNAUSTKIESTEAS	AND TEPRISSIN PETROPHYRY SAME BSONST.01.PJ	1.Amen-bv; 10-5ggevt231Auj_vj52tbg1-tw; c
	content-type: application x-awarteria-or secon	AUT VID		
	0000 00 0c 42 95 7c 5f 74 2f 68 f8 3d b0 08 0 0010 03 95 f1 77 40 00 40 06 6d 35 c0 a8 b4 f8 0020 44 08 8d 4a 00 50 7a e7 5a 6c 3f bd 43 73 0020 a3 6d 6f 38 00 00 01 01 88 0a 00 bd 6c 18	45 00		
1	Cile "Imade Mart Instrikehade Darkster 17661	Sealand: 32 Marked: 01 and 1	mer (2:02 241	Drafter Default

Gambar 20. Paket Logout dari Host Facebook

Klik kanan, dan pilih *Follow TCP String* untuk melihat detailnya, dan akan muncul seperti gambar berikut ini.



Gambar 21. Detail Paket dari Logout Facebook

Untuk melihat server enkripsi yang digunakan situs ini, maka dapat dilakukan dengan mengetik pada *filter*, sintaks berikut : *ssl contains* "*facebook*". Maka akan muncul paket data protokol SSL (*Secure Socket Layer*) yaitu protokol keamanan data yang digunakan untuk menjaga pengiriman data antar *server* dan *client*.



Gambar 22. Tampilan Filter SSL Facebook



Gambar 23. Detail Paket Protokol SSLv1 pada Facebook

3.3.1. Analisis Paket HTTPS (HTTP Secure)

Berdasarkan paket *logout* yang terekam, maka dapat dilihat bahwa paket dalam keadaan sudah terenkripsi, hal tersebut terlihat pada baris yang diseleksi yaitu *cookie*, seperti *datr* = NG51Ugn8Md-RnipEwrEEIrM2, dan facebook menulis data ID pengguna yaitu pada c_user = 100001236222422 dan selanjutnya. Facebook menggunakan jasa suatu server untuk keperluan enkripsi. Hasilnya berdasarkan gambar, diketahui bahwa server yang mememberikan tanda tangan digital dalam keperluan dekripsi, terlihat pada gambar, yaitu <u>www.verisign.com</u>.

Perbandingan antara Wireshark dan Cain

Wireshark	Cain
Untuk penggunaan	Tidak tersedia untuk
pada OS Linux,	Linux, dan tidak
dibutuhkan aplikasi	memerlukan aplikasi lain
pendukung seperti	sebagai pembantu.
Ettercap agar	
mendapat hasil	
sniffing yang lebih	
baik.	
Memiliki banyak	Fungsinya lebih ke
fungsi seperti	enkripsi kata sandi
pemeliharaan	jaringan
jaringan, mencari	
pengguna arp	
poisoning, dll.	

4. Kesimpulan

Berdasarkan dari proses pengujian yang telah dilakukan, dapat disimpulkan bahwa keamanan yang dimiliki oleh jaringan tanpa kabel atau wireless cukup rentan akan penyadapan, karena dengan melalui Cain, serangan *Man In The Middle Attack* dapat dilakukan tanpa adanya rintangan atau kemanan yang harus dilalui.

Daftar Rujukan

- [1] Witono Timotius. 2006. Linux-Based Access point Dalam Wireless LAN. (Jurnal Informatika, Vol. 2, No.2). Bandung : Sekretariat Jurnal Informatika UKM
- [2] Arifin Zaenal. 2008. Sistem Pengamanan Jaringan Wireless LAN. Yogyakarta : ANDI
- [3] Purbo, Onno.W, 1998, *TCP/IP Standar*, *Desain dan Implementasi*, PT. Elex Media Komputindo, Jakarta.
- [4] Joko I. Mumpuni dan Adisuryo Wardono. 2006. Meningkatkan Kemampuan Jaringan
- [5] Komputer. Yogyakarta : ANDI
- [6] Prakoso Samuel. 2005. Jaringan Komputer Linux Konsep Dasar, Instalasi, Aplikasi, Keamanan, dan Penerapan. Yogyakarta : ANDI
- [7] Nuraksa Makodian dan Lingga Wardhana.
 2009. Teknologi Wireless Communication dan Wireless Broadband. Yogyakarta : ANDI
- [8] Stalling W, Data and Computer Communication, Macmillan Publishing, 1985.
- [9] Hartono, Jogianto. 1999. Pengenalan omputer, dasar ilmu komputer, pemrograman, sistem informasi dan intelegnsi buatan. Yogyakarta : Penerbit ANDI
- [10] Stalling, W.Local Network, Macmillan Publishing Company, 1985
- [11] Arifin Zainal.2005.Langkah Mudah Membangun Jaringan Komputer. Yogyakarta :Penerbit ANDI
- [12] IEEEb <u>http://www.IEEE802.org/11/</u> Diakses terakhir pada tanggal 11 November 2012 , pukul 22.55 WITA
- [13] Wireless Technology
- [14] Hidayat Risanuri.2003. Wireless LAN. Yogyakarta
- [15] http://compnetworking.about.com/cs/wirel ess80211/a/aa80211standard.htm/ Diakses terkahir pada tanggal 20 November 2012, pukul 18:05 Signal Interferensi
- [16] <u>http://www.far-far-away.com/~yousif/articles/wifi-sig.php</u> Diakses terakhir pada tanggal 25 Oktober 2012, pukul 22.00 WITA.