MONITORING JARINGAN *WIRELESS* TERHADAP SERANGAN *PACKET SNIFFING* DENGAN MENGGUNAKAN IDS

Achmad Rizal Fauzi

D3 Manajemen Informatika, Fakultas Teknik, Universitas Negeri Surabaya, rizallfauzi72@gmail.com

I Made Suartana

Jurusan Teknik Informatika, Fakultas Teknik, Universitas Negeri Surabaya, madesuartana@unesa.ac.id

Abstrak

Pada zaman yang sudah maju ini banyak terdapat fasilitas *access point* secara gratis dan terbuka buat semua yang menggunakanya. *Access point* sangat rentan terhadap berbagai ancaman serangan, salah satu contoh serangan adalah dengan menggunakan *packet sniffing*, karena komunikasi yang terjadi bersifat terbuka. Diperlukan sistem keamanan dan pendeteksi yang baik agar terhindar dari serangan yang dilakukan oleh orang-orang yang tidak bertanggung jawab. Penelitian ini membahas pendeteksi serangan packet *sniffing* pada fasilitas *access point* dengan menggunakan sistem *IDS. Intrusion Detection System* (*IDS*) adalah sebuah sistem yang melakukan pengawasan terhadap *traffic* jaringan dan pengawasan terhadap kegiatan-kegiatan yang mencurigakan didalam sebuah sistem jaringan. Pada saat melakukan simulasi serangan *packet sniffing* dengan indikasi *arp spoof* menggukan *tools ettercap*. Pada protokol HTTP, *tools ettercap* mampu merekam aktivitas jaringan internet dan mampu menangkap user dan password pada saat melakukan *login* pada protokol HTTP, pada protokol HTTPS tidak dapat melakukan aktivitas internet karena protokol HTTPS memiliki *security*. Pada saat IDS snort dijalankan, maka IDS akan memonitoring jaringan internet yang sedang terhubung. Ketika menemukan kegiatan-kegiatan yang mencurigakan terutama sebuah serangan *packet sniffing* dengan indikasi *arp spoofing*, maka IDS akan memberikan *alert* berupa text "Overwrite Attack" pada PC yang sudah terinstall IDS.

Kata Kunci : *Sniffing, Intrusion Detection System, Access Point*

Abstract

In this global era there are many access point facilities for free and for all who want to use it. Access points are very vulnerable to various threats of attack, one example is an attack that used packet sniffing, it happens because of the communication is fully open for everyone. A good security and detection system is required to avoid attacks by irresponsible people. This study discusses the detection of packet sniffing attacks on the access point facility using the IDS system. Intrusion Detection System (IDS) is a system that monitors network traffic and surveillance of suspicious activities within a network system. If any suspicious activity is found related to network traffic then IDS will alert the system or network administrator. When make simulation attack of packet sniffing with *arp* spoof using tools ettercap. The protocol HTTP, tools ettercap can records internet of activites network and catch the user and password when login in HTTP protocol. HTTPS protocol can't using internet activities because HTTPS protocol has security, when IDS snort is run, it means that IDS can monitoring the internet network is being connected. When founds the suspicious activities, especially attack of packet sniffing with arp spoofing indications, this IDS can make or give some alert in text "Overwrite Attack" in the pc has been installed by IDS.

Keywords: *Sniffing, Intrusion Detection System, Access Point*.

PENDAHULUAN

Pada saat ini keamanan jaringan internet menggunakan *wireless LAN* maupun *wired local area network (LAN)* menjadi sangat penting dan patut untuk diperhatikan, jaringan yang terhubung dengan internet pada dasarnya tidak aman dan selalu dapat diekploitasi oleh para *hacker*. Pada saat data dikirim akan melewati beberapa terminal untuk sampai tujuan berarti akan memberikan kesempatan kepada pengguna lain yang tidak bertanggung jawab untuk menyadap atau mengubah data tersebut. Perancangan sistem keamanan jaringan *wireless* yang terhubung ke internet harus direncanakan dan dipahami dengan baik agar dapat melindungi sumber daya yang berada dalam jaringan tersebut secara efektif dan meminimalisir terjadinya serangan oleh para *hacker*.

Sistem keamanan jaringan *wireless* yang terhubung ke internet harus direncanakan dan dipahami dengan baik agar dapat melindungi sumber daya yang berada dalam

jaringan tersebut secara efektif. Jenis-jenis serangan yang dilakukan oleh para *hacker* antara lain, *Packet sniffer, ARP spoofing / ARP poisoning, probe, scan, Account compromise, Root compromise*, dan *Denial of service* (Dos). Salah satu ancaman yang akan di terima oleh pengguna fasilitas *access point* adalah serangan *packet sniffing*.

Packet sniffing adalah teknik pemantauan setiap paket yang melintasi jaringan, dan bagian dari perangkat lunak atau perangkat keras yang memonitor semua lalu lintas jaringan. Potensi bahaya *packet sniffing* adalah hilangnya prifasi, dan tercurinya informasi penting dan rahasia yang dimiliki oleh *user*. *ARP Spoofing* adalah. *ARP (Address Resolution Protocol) poisoning* ini adalah suatu teknik menyerang pada jaringan komputer lokal baik dengan media kabel atau *wireless*, yang memungkinkan penyerang bisa mengendus frame data pada jaringan lokal dan atau melakukan modifikasi *traffic* atau bahkan menghentikan *traffic*.

Dalam melakukan serangan packet *sniffing* banyak *tools* yang digunakan, Salah satu *tools* untuk melakukan serangan packet *sniffing* adalah *Ettercap. Ettercap* adalah sebuah *tools packet sniffer* yang dipergunakan untuk menganalisa protokol jaringan dan mangaudit keamanan jaringan. Dan memiliki kemampuan untuk memblokir lalu lintas pada jaringan *LAN*, mencuri password, dan melakukan penyadapan aktif terhadap protokol-protokol umum, *packet sniffing* juga dapat di salah gunakan oleh pihak yang tidak bertanggung jawab untuk mencuri data penting yang dimiliki oleh user yang sedang terhubung dengan *acces point*.

untuk mengatasi permasalahan perlu sebuah mekanisme keamanan jaringan untuk mendeteksi serangan *packet sniffing* dengan indikasi *ARP spoofing* pada jaringan dengan menggunakan *Intrusion Detection System (IDS)* sehingga dapat memonitoring dan memberikan alert ketika terjadi aktivitas jaringan internet yang mencurigakan.

KAJIAN PUSTAKA

Jaringan Komputer

Jaringan komputer (computer network) adalah hubungan dua buah komputer atau lebih yang bertujuan untuk melakukan pertukaran data dengan mudah. Di antaranya berbagi pemakaian perangkat lunak (software) dan perangkat keras (hardware), bahkan berbagi kekuatan pemprosesan data sehingga mempersingkat waktu pengerjaan dan meningkatkan efisiensi kerja (Kuswayatno, 2006).

Keamanan Jaringan

Keamanan jaringan adalah data-data yang berada pada perangkat keras dan perangkat lunak dalam sistem jaringan dilindungi dari tindakan-tindakan yang bersifat jahat atau merusak, modifikasi dan hal-hal yang bersifat membocorkan data ke pihak lain, untuk memastikan sistem akan berjalan secara konsisten dan handal tanpa adanya gangguan pada sistem tersebut (Binanto, 2007).

Packet Sniffing

Packet Sniffing adalah teknik pemantauan setiap paket yang melintasi jaringan. *Packet sniffing* adalah bagian dari perangkat lunak atau perangkat keras yang memonitor semua lalu lintas jaringan. Ini tidak seperti jaringan host standar yang hanya menerima lalu lintas yang dikirim khusus untuk mereka. Ancaman keamanan yang disajikan oleh penyadapan adalah kemampuan mereka untuk menangkap semua lalu lintas masuk dan keluar, termasuk password dan username atau bahan sensitif lainnya.

Untuk dapat membaca dan menganalisa setiap protokol yang melintasi jaringan, diperlukan program yang bisa membelokkan paket ke komputer *attacker*. Biasa disebut serangan *spoofing, attaker* akan bertindak sebagai *Man-In-the-Middle* (Asrodia & Patel, 2012:1).

Arp Spoofing / Poisoning

ARP (Address Resolution Protocol) poisoning ini adalah suatu teknik menyerang pada jaringan komputer lokal baik dengan media kabel atau *wireless*, yang memungkinkan penyerang bisa mengendus frame data pada jaringan lokal dan atau melakukan modifikasi *traffic* atau bahkan menghentikan *traffic*. *ARP spoofing* merupakan konsep dari serangan penyadapan diantara terhadap dua mesin yang sedang berkomunikasi atau yang disebut dengan *MITM (Man in The Middle Attack).*

Prinsip serangan *ARP poisoning* ini memanfaatkan kelemahan pada teknologi jaringan komputer itu sendiri yang menggunakan *arp broadcast. ARP* berada pada layer 2, dimana alamat pada layer dua adalah *MAC address*. Misalnya sebuah *host* (contoh: PC) yang terhubung pada sebuah *LAN* ingin menghubungi *host* lain pada *LAN* tersebut, maka dia membutuhkan inforamsi *MAC address* dari *host* tujuan (Oktavianto, 2012).

Intrusion Detection System (IDS)

Menurut Ariyus, *Intrusion detection system* dapat didefinisiakan sabagai *tools*, metode, sumber daya yang memberikan bantuan untuk melakukan identifikasi, memberikan laporan terhadap aktivitas jaringan komputer. *Intrusion Detection System* (IDS) dapat mendeteksi aktivitas yang mencurigakan dalam sebuah sistem jaringan, jadi IDS merupakam sebuah sistem

komputer yang dapat dikomobinasikan antara *hardware* dan *software* yang dapat melakukan deteksi penyusupan pada jaringan. IDS pada dasarnya adalah suatu sistem yang memiliki kemampuan untuk menganalisa data secara *realtime* dalam mendeteksi, mencatat (log) dan menghentikan penyalahgunaan dan penyerangan.

IDS merupakan *security tools* yang dapat digunakan untuk menghadapi aktivitas *hackers*. Apabila ada aktivitas yang dianggap mencurigakan di dalam jaringan maka IDS ini akan memberitahukan atau mendeteksi terhadap searangan tersebut, namun IDS tidak dapat melakukan tidakan atau pencegahan jika terjadi serangan atau penyusupan di dalam jaringan tersebut (Ariyus, 2007).

***Intrusion Prevention System* (IPS)**

Intrusion Prevention System (IPS) adalah sebuah perangkat lunak atau perangkat keras yang bekerja untuk *monitoring* trafik jaringan, mendeteksi aktivitas yang mencurigakan dan melakukan pencegahan dini terhadap penyusupan atau kejadian yang dapat membuat jaringan menjadi berjalan tidak seperti sebagaimana mestinya. IPS merupakan pendekatan yang sering digunakan untuk membangun sistem keamanan komputer, IPS mengombinasikan teknik *firewall* dan metode *intrusion detection system* (IDS) dengan sangat baik. Teknologi ini dapat digunakan untuk mencegah serangan yang akan masuk ke jaringan lokal dengan memeriksa dan mencatat semua paket data serta mengenali paket dengan sensor saat seragan teridentifikasi. Jadi IPS bertindak seperti layaknya *firewall* yang akan mengizinkan atau menghalang paket data (Raven Alder, 2007).

Snort

Snort tidak lain sebuah aplikasi atau *tools security* yang berfungsi untuk mendeteksi intrusi-intrusi jaringan (penyusupan, penyerangan, pemindaian dan beragam bentuk ancaman lainnya), sekaligus juga melakukan pencegahan. Istilah populernya, *snort* merupakan salah satu *tools Network Intrusion Prevention System* (IPS) dan *Network Intrusion Detection System* (NIDS). Dalam praktiknya, *snort* sangat handal untuk membentuk *logging* paket-paket dan analisis trafik-trafik secara *real-time* dalam jaringan-jaringan berbasis *TCP/IP*.

Snort ditulis oleh Martin Roesch bertindak sebagai pendiri dan CTO (Chief of Technical). *Snort* bukanlah sebatas protokol analisis atau sistem pendeteksi penyusupan IDS (Intrusion Detection System), melainkan sedikit gabungan dari keduanya, dan sangat berguna dalam merespon insiden-insiden penyerangan terhadap host-host jaringan.

1. Banyak dari fitur-fitur *snort* yang mirip dengan kombinasi *TCP dump/review*, tetapi snort memiliki banyak kelebihan lainya. Sebagaimana *tools ethereal* yang terkenal, snort tersedia bebas dalam bentuk *source code* di bawah lisensi *GNU Genaral Public License*, untuk kebanyankan varian dan distro *linux/unix*, dan juga sistem-sistem *windows* (Rahmat, 2010).

Preprocessors

Preprocessors Snort terbagi menjadi dua kategori, yaitu dapat digunakan untuk memeriksa paket yang mencurigakan atau memodifikasi paket sehingga mesin deteksi dapat mendeteksi dengan benar. Sejumlah serangan tidak dapat dideteksi melalui mesin pendeteksi, jadi diperlukan alert khusus untuk *preprocessors* mendeteksi aktivitas yang mencurigakan. Jenis preprocessors ini sangat diperlukan dalam menemukan serangan berbasis non-tanda tangan. *Preprocessors* lainnya bertanggung jawab untuk menormalkan lalu lintas sehingga mesin deteksi dapat mencocokkan alert secara akurat. *Preprocessors* ini mengalahkan serangan yang berusaha menghindari mesin deteksi *Snort* dengan memanipulasi pola lalu lintas.

Paket siklus *Snort* melalui setiap *preprocessors* untuk menemukan serangan yang memerlukan lebih dari satu *preprocessors* untuk mendeteksi aktivitas yang mencurigakan. Jika Snort berhenti mengecek atribut paket yang mencurigakan setelah memasang peringatan melalui *preprocessors*, penyerang bisa menggunakan kekurangan ini untuk menyembunyikan lalu lintas dari *Snort*. Misalkan paket sengaja dikodekan/enkripsi, serangan mengeksploitasi jarak jauh berbahaya dengan cara yang akan memicu peringatan prioritas rendah dari *preprocessors*. Jika pemrosesan diasumsikan selesai pada saat ini dan paket tidak lagi disimpan melalui *preprocessors*, serangan eksploitasi jarak jauh hanya akan mendaftarkan sinyal pengkodean. Eksploitasi jarak jauh akan luput dari perhatian *Snort*, mengaburkan sifat sebenarnya dari lalu lintas. Parameter *preprocessors* dikonfigurasi dan diatur melalui file snort.conf. File snort.conf yang sama memungkinkan Anda menambahkan atau menghapus *preprocessors* sesuai keinginan pengguna (Koziol, 2003).

***Arp* Prepropcessors**

Arp spoof adalah *preprocessors* yang dirancang untuk mendeteksi jalannya *Address Resolution Protocol* (*ARP*). *arp* digunakan pada jaringan *ethernet* untuk memetakan alamat IP ke alamat MAC. Untuk mengurangi jumlah siaran *arp* pada jaringan modern, sistem operasi perangkat yang terhubung menyimpan cache pemetaan

arp. Saat perangkat menerima balasan *arp*, maka *cache* pada *arp* akan diperbarui dengan pemetaan alamat IP ke MAC yang baru apakah perangkat tersebut mengirim permintaan *arp* atau tidak. Berbagai serangan melibatkan *arp*. *Spoofing ARP* dilakukan dengan menyusun *arp request* dan *reply* paket. Paket balasan *arp* yang ditangguhkan disimpan di *cache arp* dari perangkat penerima meskipun perangkat tidak mengirim permintaan.

Jenis serangan *arp spoof* lainnya adalah serangan *arp* menimpa serangan. Serangan tersebut bekerja dengan mengirimkan paket *arp* yang diterima oleh perangkat untuk alamat antarmuka perangkat itu sendiri tetapi dengan alamat MAC yang berbeda. Ini akan menimpa alamat MAC perangkat itu sendiri di *cache arp* dengan permintaan *arp* yang berbahaya. Hal ini menyebabkan perangkat tidak dapat mengirim dan menerima paket *arp*. Pada gilirannya, ini menyebabkan perangkat dan perangkat lain yang bergantung padanya agar komunikasi tidak dapat mengirim paket satu sama lain. Karena *arp* adalah protokol Layer 2, *arp spoof* hanya mendeteksi serangan yang terjadi pada segmen fisik yang sama seperti sensor *Snort*. *arp spoof* memiliki dua pilihan konfigurasi, yaitu:

1. host IP address host MAC address

Setiap perangkat yang ingin di monitor dengan *arp* spoof harus ditentukan dengan pemetaan alamat Ip dan MAC miliknya sendiri. Masing-masing perangkat terdaftar pada baris baru di file snort.conf. Setiap kali pemetaan berubah, Anda harus mengkonfigurasi ulang file tersebut. Perangkat yang mendapatkan alamat IP mereka melalui DHCP harus dikonversi ke IP statis sebelum *ARP*spoof diaktifkan (Koziol, 2003).

2. Unicast

Pilihan ini akan memungkinkan deteksi serangan *Arp unicast*. Sebagian besar permintaan *arp* yang *valid* dikirim ke alamat *broadcast*. Permintaan *arp* yang dikirim ke alamat *Unicast* seringkali merupakan tanda serangan yang dirancang untuk memodifikasi *cache arp*. Pilihan ini dinonaktifkan secara *default*, namun dapat diaktifkan jika terdapat penyalahgunaan *Arp* yang serius (Koziol, 2003).

METODE

Analisa Sistem

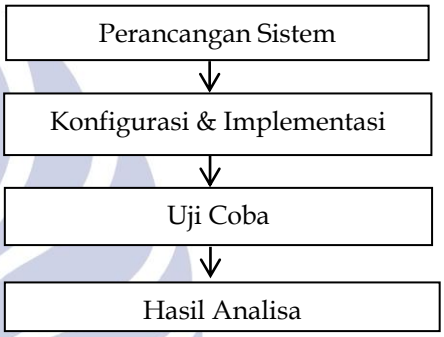
IDS merupakan *security tools* yang dapat digunakan untuk menghadapi aktivitas *hackers*. Apabila ada aktivitas yang dianggap mencurigakan di dalam jaringan maka IDS ini akan memberitahukan atau mendeteksi terhadap searangan tersebut, namun IDS tidak dapat melakukan tidakan atau pencegahan jika terjadi serangan atau penyusupan di dalam jaringan tersebut. Dalam implementasi proyek tugas akhir ini akan menggunakan 1 buah *attacker* (ubuntu 16.10), 1 buah pendeteksi (ubuntu 16.04), dan 1 buah user (windows 8).

Penulis berharap dengan menggunakan ids dapat memonitoring jaringan internet pada penggunaan access point di tempat umum, sehingga dapat mempermudah pekerjaan bagi para administrator jika terjadinya serangan dari orang yang tidak bertanggung jawab.

ditentukan oleh perusahaan, baik infrastruktur, platform, maupun aplikasi yang ada. (Alex Budiyanto, 2012)

Desain Sistem

Desain Sistem yang dilakukan penulis seperti gambar 1 berikut.



Gambar 1. Desain Sistem

Penjelasan masing-masing desain sistem dalam metode penelitian ini adalah sebagai berikut.

1. Perancangan Sistem

Tahap ini merupakan tahap awal yang akan dilakukan untuk melakukan penelitian tentang keamanan jaringan pada fasilitas internet wifi terhadap serangan *packet sniffing* dengan menggunakan *ids.*

2. Konfigurasi & Implementasi

Install aplikasi Ettercap pada *linux* yang digunakan untuk melakukan serangan *packet sniffing*, setelah melakukan instalasi penulis melakukan konfigurasi terhadap aplikasi Ettercap dan install juga aplikasi / *tools ids* yang digunakan untuk melakukan pendeteksi adanya serangan packet sniffing, dan juga penulis juga membuat rule-rule tertentu agar dapat mendeteksi serangan packet sniffing dengan indikasi *arp spoofing*.

3. Uji Coba

Pertama user akan terhubung ke access point yang sudah tersedia, lalu PC penyerang akan melakukan serangan *packet sniffing* terhadap access point, maka PC pendeteksi akan mendeteksi adanya serangan *packet sniffing* dengan indikasi *arp spoofing.*

4. Hasil Analisa

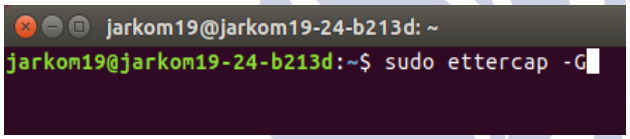
Penulis akan menganalisa hasil uji coba serangan *packet sniffing* dan mendeteksi serangan dengan menggunakan *ids*.

HASIL DAN PEMBAHASAN

Piranti yang diperlukan dalam membuat proyek yang berjudul monitoring jaringan wireless terhadap serangan packet dengan menggunkana IDS, memerlukan 2 pc sebagai attacker dan pendeteksi, OS yang digunakan pada 2 pc tersebut menggunakan ubuntu 16.04

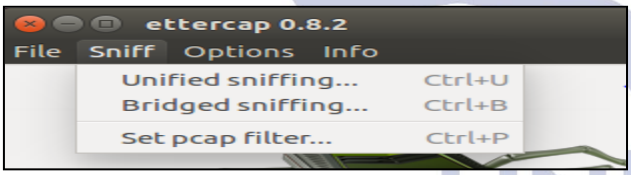
Gambar 2 Tampilan dekstop pada ubuntu 16.04LTS

Pada gambar dibawah ini akan melakukan pengujian aplikasi ettercap dapat menangkap aktivitas jaringan internet yang sudah tersedia, user akan melakukan aktivitas internet dengan menggunakan protokol HTTP dan protokol HTTPS, untuk langkah pertama menjalankan aplikasi ettercap.



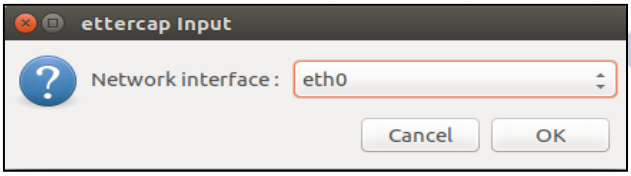
Gambar 3 Menjalankan aplikasi *ettercap*

Langkah selanjutnya pilih pada bagian sniif kemudian klik kiri pada pilihan *unified sniffing*.



Gambar 4 Tampilan menu *ettercap*

Pada langkah selanjutnya Pilih interface eth0, setelah itu tekan tombol OK

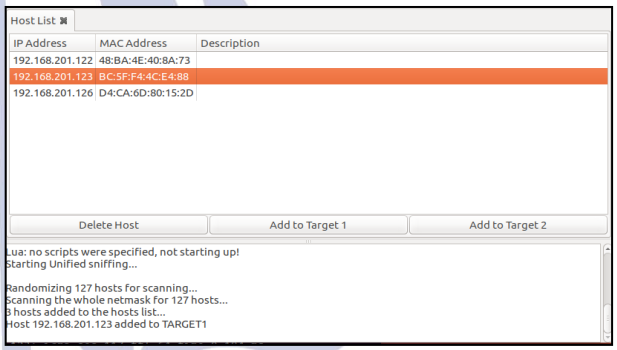


Gambar 5 Pemilihan *interface*

Setalah memilih *interface*, langkah selanjutnya pilih scan host yang berfungsi menscan *host* yang terhubung pada *access point*.

Gambar 6 Pilihan pada *menu host*

Setelah proses scan host selesai, kemudian pilih *host list*, maka akan menampilkan *ip address* dan *mac address* yang terhubung pada *acces point*. Langkah selanjutnya pilih *ip address* mana yang akan diserang, pada gambar dibawah ini menunjukkan *ip address* yang diserang adalah ip 192.168.201.23 sebagai target 1.

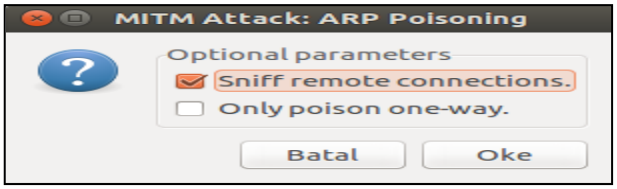


Gambar 7 Tampilan pada *host list*

Setelah menentukan ip mana yang akan diserang, langkah selanjutnya pilih Mitm kemudian pilih ARP *poisoning*.

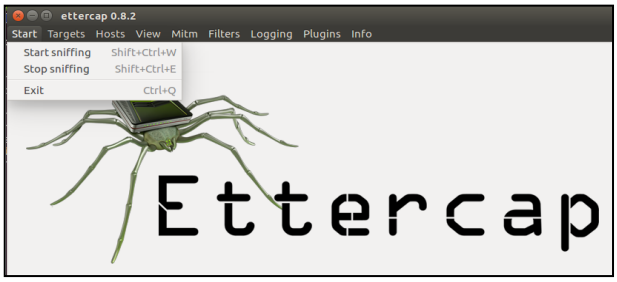
Gambar 8 Tampilan menu mitm

Kemudian centang pada *sniff remote connections*, lalu klik oke.



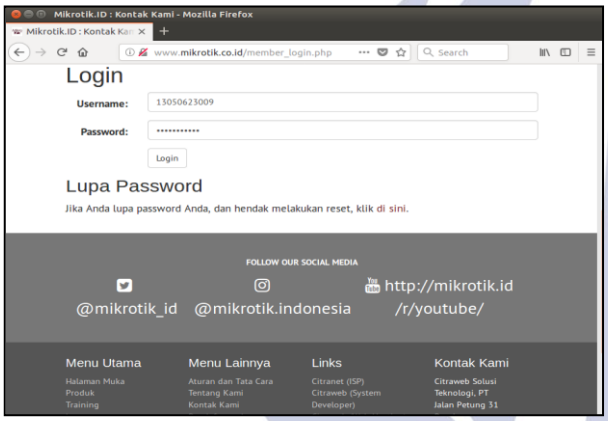
Gambar 9 Tampilan optional parameters

Pilih Start pada menu Ettercap kemudian pilih Start *sniffing*.



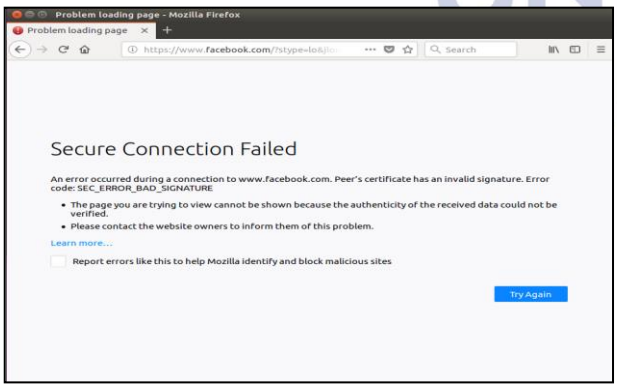
Gambar 10 Start *sniffing*

Setalah menjalankan aplikasi ettercap, ujicoba pertama dan kedua user akan melakukan aktivitas internet dengan menggunakan protokol HTTP pada halaman *login* mikrotik http://www.mikrotik.co.id/member_login.php



Gambar 11 Halaman *login* mikrotik

Ketika user menggunakan protokol HTTPS untuk melakukan aktivitas internet, pada gambar dibawah ini user membuka web pada halaman login facebook https://www.Facebook.com akan tetapi pada saat membuka halaman *login* Facebook tidak bisa dikarenakan HTTPS memiliki keamanan sendiri yang bagus.



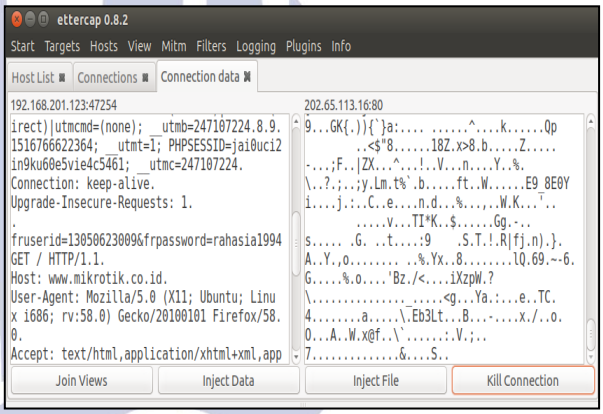
Gambar 12 Tampilan *login* facebook

Setelah melakukan serangan pada protokol HTTP dan HTTPS langkah selanjutnya menjalankan snort apakah dapat memonitoring serangan packet sniffing dengan indikasi arp spoof atau tidak, untuk menjalankan snort gunakan perintah sebagai berikut:

```
Sudo snort –A console –i wlan0 –u
snort –g snort –c
/etc/snort/snort.conf
```

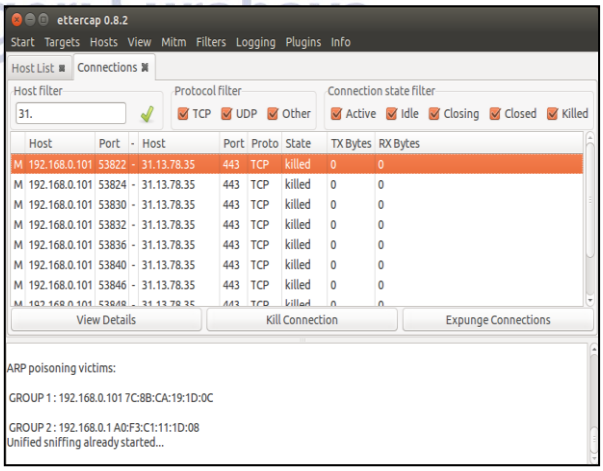
Gambar 13 Menjalankan snort

Pada bagian *connection* data dapat dilihat terdapat *user* dan *password* setelah melakukan *login* pada halaman mikrotik.



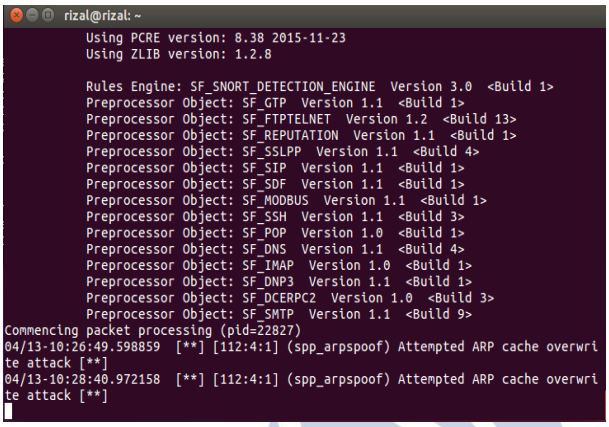
Gambar 14 Tampilan connection data

Pada hasil ketika user melakukan aktivitas internet dengan menggunakan protokol HTTPS, pada gambar dibawah ini menunjukkan hasil aktivitas, akan tetapi pada saat di klik data aktivitas internet pada halaman web *facebook* terenkripsi, ini adalah keunggulan dari protokol HTTPS yang sangat aman digunakan untuk melakukan aktivitas internet.



Gambar 15 Tampilan connection data facebook

Pada gambar dibawah ini menunjukkan apakah *snort* dapat memonitoring atau tidak serangan *packet sniffing* dengan indikasi *arp spoof*, pada gambar dibawah ini menunjukkan bahwa snort mampu memonitoring serangan yang sedang berjalan pada *access point*, *snort* memberikan alert berupa (Attempted ARP cache overwrite attack) alert tersebut menunjukkan bahwa jaringan internet yang tersedia sedang ada orang yang tidak bertanggung jawab melakukan serangan berupa *packet sniffing* dengan indikasi *arp spoof*.



Gambar 16 Hasil monitoring snort

KESIMPULAN DAN SARAN

Simpulan

Berikut adalah kesimpulan yang di dapat dari hasil implementasi yang telah dibuat:

1. *Tools ettercap* dapat merekam dengan baik ketika user melakukan aktivitas internet menggunakan protokol HTTP, berbeda ketika *user* menggunakan HTTPS dimana data aktivitas yang terekam pada *tools ettercap* akan terenkripsi.
2. Berdasarkan hasil pengujian bahwa pada saat *user* menggunakan protokol HTTP untuk melakukan aktivitas internet maka tidak adanya gangguan dalam jaringan *access point*, berbeda pada saat menggunakan protokol HTTPS dikarenakan HTTPS sudah memiliki *security* sendiri sehingga ketika adanya serangan pada jaringan internet maka secara otomatis halaman *web* yang menggunakan protokol HTTPS akan muncul peringatan (Secure Connection Failed)
3. Dalam mendeteksi serangan *packet sniffing* dengan indikasi *arp spoof* menggunakan *snort*. Agar *snort* dapat memonitoring serangan *packet sniffing* diperlukan konfigurasi pada *snort.conf* dan menggunakan *rule preprocessor*, ketika terjadinya serangan *packet sniffing* pada jaringan internet maka secara otomatis *snort* akan memberikan *alert* berupa (Attempted ARP cache overwrite attack).

Saran

Diharapkan dapat sebagai referensi dan bahan masukan ilmu pengetahuan baru kepada pembaca tentang *ettercap* dan *snort*. Dan diharapkan pada penelitian selanjutnya mampu melakukan penelitian dengan menggunakan *tools* yang berbeda dan mampu menambahkan tindakan pada saat terjadinya serangan pada jaringan internet. Sehingga kedepanya jaringan internet pada tempat umum bisa aman dengan adanya monitoring jaringan internet dan tindakan terhadap serangan yang terjadi pada jaringan internet.

DAFTAR PUSTAKA

Intrusion Detection, Second Edition. Published by Rockland.

Ariyus, Doni. 2007. *Intrusion Detection System*. Yogyakarta: Andi.

Asrodia, Pallavi & Hemlata Patel. 2012. *Analysis of Various Packet Sniffing Tools for Network Monitoring and Analysis*. *(Online).* (www.researchtrend.net, diakses 3 Januari 2017).

Iwan, Binanto. (2007). *Membangun Jaringan Komputer Praktis Sehari-hari*. Yogyakarta: Graha Ilmu.

Koziol, Jack. 2003. *Intrusion Detection with Snort.* Published by Sams.

Kuswayatno, Lia. 2006. *Mahir Berkomputer*. Bandung: Grafindo Media Pratama.

Oktavianto, Digit. 2012. *Mencegah ARP Spoofing dan ARP Poisoning di Linux. (Online).* (http://digitoktavianto.web.id/mencegah-*arp*-spoofing-dan-*arp*poisoning-di-linux.html, diakses 3 Januari 2017).

Rafiudin, Rahmat. 2010. *Mengganyang Hacker dengan Snort:Andi*. Yogyakarta: OFFSET.