**BAB I**

**PENDAHULUAN**

1. **Latar Belakang**

*Address Resolution Protocol (ARP)* memiliki tanggung jawab untuk mencari *Media Access Control (MAC) Address* dari setiap komputer *(host)* yang akan berkomunikasi (pertukaran data) melalui jaringan *Local Area Network (LAN)* dengan memanfaatkan *Internet Protocol (IP) Address*, khususnya *IP Address Version 4* yang telah didapat saat sebuah *host* berhasil melakukan koneksi ke jaringan. *MAC Address* yang telah didapatkan dari *ARP* akan disimpan di dalam *ARP Cache/ARP Table* dengan tujuan agar jaringan tidak menjadi sibuk karena penggunaan *ARP* secara terus-menerus ketika setiap *host* ingin berkomunikasi dengan *host* lainnya. Dalam *Open System Interconnection (OSI)* protokol ini sering digambarkan sebagai yang berada antara lapisan *(*l*ayer)* ke 2 dan 3 dimana juga dienkaptulasi oleh protokol *layer* 2.

Protokol ini sangat sering dimanfaatkan oleh penyerang *(hacker/attacker)* untuk melakukan serangan *Man In The Middle (MITM)* dengan berbagai tujuan. Ketika *attacker* mengirimkan paket *ARP Reply* yang telah dimodifikasi sedemikian rupa dengan tujuan *host* targetmaka *ARP Cache* dari targetakan ter-*update* sesuai dengan apa yang diinginkan oleh *attacker*. Setiap *host* yang ingin berkomunikasi dengan *host* lainnya menggunakan  *MAC Addres* agar setiap *frame* datadapat diteruskan ke tujuan dan diletakan di atas media transmisi (kabel atau radio) dan diproses oleh *Network Interface Card (NIC)* dimana *MAC Address* tersebut tersimpan di dalam *ARP Cache* yang telah ter-*update.* Hal ini dapat menyebabkan paket yang dikirimkan tidak sampai pada *host* tujuan atau justru dapat diterima oleh *attacker*. Akibat dari serangan ini salah satunya adalah hilangnya kerahasiaan dari data yang dikirimkan melalui jaringan ini seperti *password, email,* dan lain sebagainya.

Serangan menggunakan *ARP* ini dapat dideteksi dengan menggunakan Jaringan Saraf Tiruan (JST) dengan melakukan normalisasi terhadap paket-packet *APR* yang berhasil di-*capture* agar dapat menjadi *input*-an bagi arsitektur Jaringan Saraf Tiruan yang telah dibuat. Jaringan ini dapat digunakan untuk memodelkan hubungan yang kompleks antara *input* dan *output* untuk menemukan pola-pola data. Salah satu algoritma/metode untuk melakukan pembelajaran terarah *(supervised learing)* pada Jaringan Saraf Tiruan dalam mencari beban (*weight*) pada setiap *neuron* yang menghasilkan nilai kesalahan seminimal mungkin adalah Perambatan Mundur *(Backpropagation)*. Oleh karena itu peneliti menggunakan metode untuk meningkatkan keakuratan dari hasil deteksi yang dilakukan oleh Jaringan Saraf Tiruan.

1. **Rumusan Masalah**

Berdasarkan latar belakang diatas dapat dirumuskan beberapa masalah dalam peneliatan “Deteksi Serangan Pada ARP (Address Resolution Protocol) Dengan Menggunakan Jaringan Saraf Tiruan”, yaitu:

* Bagaimana setiap paket yang melewati *Network Interface Card (NIC)* dapat di-*capture*?
* Bagaimana mengolah paket-paket yang berhasil di-*capture* agar dapat dilakukan normalisasi data yang akan digunakan sebagain *input* untuk Jaringan Saraf Tiruan?
* Bagaimana meningkatkan presentasi pengenalan paket-paket yang dapat membayakan?

1. **Batasan Masalah**

Mengingat dengan banyaknya perkembangan masalah yang bisa ditemukan pada penelitian ini, maka perlu adanya batasan-batasan masalah yang jelas mengenai apa yang dibuat dan diselesaikan. Adapun batasan-batasan masalah pada penelitian ini, sebagai beribut:

* Mencari data paket hasil *capture* yang akan menjadi *input-*an.
* Menganalisa *input-*an berupa paket data.
* Memastikan komputer yang digunakan sedang mendapatkan serangan atau tidak.

1. **Tujuan Penelitian**

Adapun tujuan dari penelitian ini adalah untuk membuat sistem yang dapat melakukan pengecekan paket-paket *ARP* sehingga dapat menciptakan sebuah sistem yang dapat mendeteksi serangan *ARP Spoofing*.

1. **Manfaat Penelitian**

Diharapkan penelitian ini dapat memberikan manfaat, diantaranya:

* Penelitian ini diharapkan dapat memberikan manfaat secara teoritis, sekurang-kurangnya dapat berguna sebagai sumbangan pemikiran bagi dunia pendidikan.
* Menambah wawasan penulis khususnya mengenai keamanan jaringan komputer.
* Sebagai referensi dalam ilmu pendidikan sehingga dapat memeperkaya dan menambah wawasan khususnya menyenai keamanan jaringan komputer.
* Sistem dapat mengurangi tingkat penyalahgunaan jaringan *LAN* baik untuk pencurian *password*, *email*, manipulasi paket data dan sebagainya.
* Penelitian ini juga diharapkan dapat bermanfaat bagi penelitian berikutnya.

**BAB II**

**KAJIAN PUSTAKA DAN TEORI**

1. **Kajian Hasil Penelitian**

Penelitian oleh Vinay dan Rahman (2015), dengan judul *“ARP Spoof Detection System using ICMP Protocol: An Active Approach”* menggunakan teknik deteksi aktif dengan cara meng-capture paket *ARP* dan menyimpannya paket pertama ke dalam *database* *(IP* dan *MAC Address)*. Ketika hasil capture paket *ARP* berikutnya telah ada (telah disimpan ke *database* sebelumnya) maka paket tersebut dinyatakan aman (*IP* dan *MAC Address* sama), namun jika yang sama hanya *IP Address* yang sama maka *IP* tersebut akan digunakan untuk melakukan pengiriman paket *ICMP (Internet Control Messaege Protocol)*. Jika *IP packet routing (IP forwarding)* pada penyerang diaktifkan maka paket yang tadi dikirimkan ke penyerang akan diteruskan kembali oleh penyerang sesuai dengan *IP Address* tujuan. Dari paket tersebut dapat dilakukan pencocokan paket pada *layer* 2 dengan paket *ARP* hasil *capture* yang sebelumnya untuk memastikan apakah paket *ARP* tersebut aman atau tidak. Namun hal ini dapat diatasi oleh penyerang dengan membuat *firewall* untuk meblokir setiap paket *ICMP* yang masuk. Oleh karena itu peneliti tidak hanya menggunakan paket *ICMP* sebuagai parameter untuk deteksi namun juga berbagai macam parameter lainnya yang masih mungkin dapat digunakan.

Kaur (2013) penah melakukan penelitian dengan judul *“Detection and Prevention of ARP Cache Poisoning”* untuk mendeteksi dan mengatasi serangan *ARP Spoofing (ARP Cache Poisoning)* dengan mendeteksi paket-paket yang mencurigakan dan ketika telah dipastikan ada yang melakukan serangan segera diambil tindakan dengan mengirimkan paket *ARP Request* ke *gateway* dengan tujuan untuk memperbaharui *ARP Cache*. Selain itu digunakan juga *ICMP* (*Internet Control Message Protocol*) untuk melakukan pengecekan apakah penyerang meng-aktifkan fungsi *IP Forwarding (IP Routing)* untuk meneruskan paket *IP* pada tujuan. Berdasarkan penelitian tersebut peneliti akan menambakan fitur untuk melakukan penyimpanan hasil *capture* paket pada format yang umum digunakan seperti pcap dan pcapng agar lebih mudah untuk dianalisis oleh peneliti lain.

Srinath dkk (2015) telah melakukan penelitian dengan judul *“Detection and Prevention of ARP Spoofing using Centralized Server”* dengan menggunakan tiga model untuk mengatasi serangan *ARP Spoofing* yaitu model perspektif komputer (*host*), perspektif *server*, dan otentikasi. Dimodel pertama setiap komputer mengirimkan informasi yang didapat setelah terhubung ke jaringan melalui *DHCP* ke *server* dan tugas *server* adalah menyimpan informasi tersebut ke *database* sekaligus melakukan pengecekan informasi. Informasi yang disimpan di *database* dapat ditampilkan dengan menggunakan diagram agar mempermudah pembacaan. Sistem yang peneliti buat hanya digunakan di sisi *client* dengan alasan kebiasaan pengguna jaringan seperti *wifi* yang selalu berpindah-pindah (tidak hanya menggunakan satu jaringan).

**Table 2.1** Perbandingan Tinjauan Pustaka

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Judul | Penulis | Motode | Kesimpulan |
| 1 | *ARP Spoof Detection System using ICMP Protocol: An Active Approach* | Vinay K. R. dan T. R. Mahibur Rahman | *ICMP Modul* | Tiknik ini juga dapat mendeteksi IP dan MAC Address yang asli (Correct Address) selain itu teknik ini cukup simple dan effisien. |
| 2 | *Detection and Prevention of ARP Cache Poisoning* | Inderjeet Kaur | *ARP dan ICMP* | Metode ini cukup efisian untuk mendeteksi dan mengatasi ARP Cache Poisoning. |
| 3 | *Detection and Prevention of ARP Spoofing using Centralized Server* | D. Srinath, S. Panimalar, A. Jerrin Simla dan J. Deepa | *Centralized Server* | Metode ini cukup baik digunakan untuk mengatasi ARP Spoofing selain itu dapat pula digunakan untuk mengatasi IP Spoofing. |

Seperti terlihat pada table 2.1. perbedaan dari ketiga referensi dengan judul yang diangkat oleh penulis terletak pada metode yang digunakan, masing-masing metode memiliki keunggulannya masing-masing. Peneliti akan menggunakan beberapa keunggulan dari masing-masing referensi dan menambahkan beberapa metode untuk meingkatkan kemampuan dari sistem pendeteksi serangan pada penelitian ini.

1. **Dasar Teori**
   1. *Intrusion Detection System* (IDS)

Intrusion Detection System merupakan sebuah sistem yang dapat digunakan untuk melakukan deteksi terhadap aktivitas yang mencurigakan dalam sebuah sistem atau jaringan yang dapat menggangu konfidensialitas, integritas dan ketersediaan data. IDS dapat melakukan inspeksi terhadap lalu lintas *inbound* dan *outbound* dalam sebuah sistem atau jaringan, melakukan analisis dan mencari bukti dari percobaan intrusi.

* 1. Protokol

Protokol pada jaringan komputer merupakan sebuah prosedur atau aturan yang harus disetujui secara bersama oleh perangkat yang akan berkomunikasi. Banyaknya protokol yang berbeda pada jaringan mengakibatkan sulitnya komunikasi antar perangkat yang terkoneksi melalui jaringan.

Salah satu model arsitektur yang banyak digunakan adalah OSI (*Open System Interconnection*) yang berupaya membentuk standar umum jaringan komputer untuk menunjang interoperatibilitas antar pemasok (*vendor*) dari yang berbeda. OSI memiliki 7 lapisan/*layer* yang setiap lapisan memiliki fungsinya masing-masing. Menurut Sugeng dan Putri fungsi dari masing-masing lapisan/*layer* yang terdapat pada OSI sebagai berikut:

1. Lapisan Fisik (*Physical Layer*), berfungsi dalam mengiriman *raw* bit ke kanal komunikasi. Masalah-masalah yang harus diperhatikan adalah masalah desain (Jika dikirim bit 1 harus diartikan bit 1 disisi penerima), masalah debain ini ditemukan ada hubungannya dengan mekanika, kelistrikan, prosedur *interface*, dan medium transmisi fisik yang berada di lapisan fisik.

2. Lapisan Jalur Data (*Data Link Layer*), tugas utamanya sebagai fasilitas transmisi *raw* data dan mentransfirmasikan data tersebut ke saluran yang bebas dari kesalahan transmisi. Dimungkinnya melalukan pemecahan data input menjadi sejumlah data *frame* (biasanya jumlahnya ratusan atau ribuan byte). Selanjutnya *frame* tersebut dikirim secara perurutan, dan memproses *acknowledgment frame* yang dikirim kembali oleh penerima. Penambahan bit-bit khusus diawal dan diakhir data guna pengenalan *frame* merupakan bagian pekerjaannya. Jika terjadi *noise* dan *frame* rusak *frame* dikirim ulang, tapi akibatnya akan terjadi duplikasi *frame* jika  *acknowledgment frame* hilang.

3. Lapisan Jaringan (*Network Layer*)*,* berfungsi sebagai pengendalian operasi *subnet*. Masalah desain yang penting adalah menentukan *route* pengiriman *packet* dari sumber ke tujuannya. Desain *route* dapat berupa statik atau dinamik. Masalah pengendalian kemacetan (*bottlenect*) merupakan tugasnya. Pada jaringan *broadcast,* masalah penentuan *route* hal yang sederhana, lapisan jaringan bisa tidak ada atau tidak diperlukan.

4. Lapisan Transport (*Transport Layer*), fungsi dasarnya adalah menerima data dari Lapisan Sesi, bila perlu memecah data menjadi bagian-bagian yang lebih kecil, meneruskan potongan ke lapisan jaringan dan menjamin seluruh potongan data sampai dengan benar disisi lainnya. Harus dilaksanakan secara efisien. Tujuan lainnya adalah melindungi seluruh lapisan diatasnya dari perubahan teknologi perangkat keras yang mungkin timbul. Bila diperlukan *throughput* yang tinggi, maka lapisa *transport* hubungan jaringan yang banyak, tetapi dapat pula menggabungkan beberapa hubungan *transport* ke hubungan jaringan yang sama. Penentuan jenis layanan (yang populer adalah saluran *error-free point tot point*) merupakan tugasnya pula. Merupakan *layer end-to-end* sejati dari sumber ke tujuan. Banyak *host* diprogram dengan *multiprogrammed* (banyak hubungan yang masuk dan meninggalkan *host* untuk menyatakan pesan mana). TH adalah tempat informasi tersebut ditempatkan. Pengendalian aliran (*Flow Control*) adalah merupakan tugasnya agar tidak membanjiri *host* yang lambat.

5. Lapisan Sesi (*Session Layer*), mengizinkan para pengguna untuk menetapkan *session* di antara mereka. Sebuah *session* digunakan untuk memungkinkan seseorang pengguna lelakukan *log* ke dalam suatu *remote time sharing system* atau memindahkan suatu *file* dari satu mesin ke mesin yang lain. Jadi tugasnya adalah pengendalian dialog. Funsi lainnya dalah manajemen *token (token management)*, sinkronisasi (*synchronization*), penyisipan *checkpoint* diperlukan jika akan mengulangi pengiriman akibat terjadinya *crash* sehingga tidak perlu seluruh data diulang pengirimannya.

6. Lapisan Presentasi (*Presentation Layer*),melakukan fungsi tertentu yang sering diminta untuk menjamin penemuan sebuah penyelesaian umum bagi masalah tertentu. Lapisan Presentasi tidak mengizinkan pengguna untuk menyelesaikan sendiri suatu masalah. Lapisan Presentasi memperhatikan *sytax* dan semantik informasi yang dikirimkan. Contoh layanannya adalah pengodean data (*data encoding*).

7. Lapisan Aplikasi (*Appilcation Layer*), tugasnya melayani *remote* terminal. Lapisan aplikasi terdiri dari bermacam-macam protocol yang bisa digunakan. Diperlukan adanya terminal virtual jaringan (*network virtual terminal*) sebelum suatu editor *remote* digunakan. Fungsi lainnya adalah pemindahan … (biasanya satu sistem ke sistem lain mempunyai konvensi yang berbeda). Tugasnya seperti: E-mail, Telnet, FTP, WWW dan lain sebagainya.

* 1. ARP

Menurut X, ARP (Address Resolution Protocol) adalah protokol yang bertugas untuk menemukan *hardware address* suatu host dengan alamat IP tertentu.

|  |  |  |
| --- | --- | --- |
| Octet Offset | 0 | 1 |
| 0 | Hardware type | |
| 2 | Protocol type | |
| 4 | Hardware Address Length | Protocol Address Length |
| 6 | Operation | |
| 8 | Sender Hardware Address | |
| 10 |
| 12 |
| 14 | Sender Protocol Address | |
| 16 |
| 18 | Target Hardware Address | |
| 20 |
| 22 |
| 24 | Target Protocol Address | |
| 26 |

* 1. Ethernet

Menurut X, Pada awalnya Ethernet didesain untuk dijalankan di atas kabel koaksial pada kecepatan maksumum 10 Mbps. Sekarang Ethernet berjalan pada kabel koaksial *thin-wide* (10base2) dan *unshielded twisted-pair (UTP) telephone wiring* (10base3). *Device* pada *network-PC, workstation, printer, server,* dll secara fisik terhubung ke kabel tunggal yang dikenal sebagai *bus.*

Pada perkembangan berikutnya, muncul teknologi *Switch Ethernet,* untuk menghindari *problem* tabrakan paket. Sebuah *Switch Ethernet* menggantikan pengabelan *hub.* Berikutnya ada *Fast Ethernet,* yang membesarkan *bandwith* LAN dari 10 Mbps menjadi 100 Mbps. Ia menggunakan 2 standar: Gigabit 100base-I (IEEE 802.3u) dan Gigabit 100VG-AnyLAN (IEEE 803.12).

* 1. ICMP

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Offset | Octet | 0 | | 1 | | 2 | | 3 | |
| Octet | Bit | 0 | 4 | 8 | 12 | 16 | 20 | 24 | 28 |
| 0 | 0 | Type | | Code | | Checksum | | | |
| 4 | 32 | Rest of Header | | | | | | | |

* 1. IP Address

Menurut X, IP (Internet Protocol) *address* atau alamat IP yang bahasa awamnya bisa disebut dengan kode pengenal komputer pada jaringan merupaan komponen vital pada internet, karena tanpa alamat IP sesorang tidak akan dapat terhubung ke *internet.* Setiap komputer yang terhubung ke *internet* setidaknya harus memiliki satu buah alamat IP pada setiap perangkat yang terhubung ke *internet* dan alamat IP itu sendiri harus unik karena tidak boleh ada komputer/*server/*perangkat jaringan lainnya yang menggunakan alamat IP yang sama di *Internet.*

* 1. IPv4

Menurut X, alamat IP (Ipv4) pada awalnya adalah sederetan bilangan biner sepanjang 32 bit yang dipakai untuk mengidentifikasikan *host* pada jaringan. Alamat IP ini diberikan secara unik pada masing-masing komputer/*host* yang terhubung ke *internet.* Prinsip kerjanya adalah paket yang membawa data dimuati alamat IP dari komputer pengirim data kepada alamat IP pada komputer yang akan dituju, kemudian data tersebut dikirim ke jaringan. Paket ini kemudian dikirim dari *router* ke *router* dengan berpedoman pada alamat IP tersebut menuju ke komputer yang dituju. Seluruh komputer/*host* yang tersambung ke iinternet, dibedakan hanya berdasarkan alamat IP untuk setiap komputer yang terhubung ke jaringan *internet.*

IPv4 terdiri dari 14 *field*, namun satu *field* terakhir hanya bersifat  *optional*. Berikut format dari paket Ipv4.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Offsets | Octet | 0 | | | | 1 | | | | 2 | | | | | 3 | | | |
| Octet | Bit | 0 | 2 | 4 | 6 | 8 | 10 | 12 | 14 | 16 | 18 | | 20 | 22 | 24 | 26 | 28 | 30 |
| 0 | 0 | Version | | IHL | | DSCP | | | E | Total Length | | | | | | | | |
| 4 | 32 | Identification | | | | | | | | Flags | | Fragment Offset | | | | | | |
| 8 | 64 | Time to Live | | | | Protocol | | | | Header Checksum | | | | | | | | |
| 12 | 96 | Source IP Address | | | | | | | | | | | | | | | | |
| 16 | 128 | Destinatin IP Address | | | | | | | | | | | | | | | | |
| 20 | 160 | Options (if IHL > 5) | | | | | | | | | | | | | | | | |
| 24 | 192 |
| 28 | 224 |
| 32 | 256 |

* 1. Sniffing & Spoofing

Sniffing adalah proses penyadapan paket pada jaringan dengan menggunakan sebuah aplikasi yang biasa disebut *Network Analyzer*. Aplikasi ini menangkap tiap-tiap paket dan dapat juga menguraikan paket tersebut berdasarkan RFC (Request of Comments). Sedangkan spoofing merupakan proses pemalsuan paket-paket jaringan yang dapat mendukung proses sniffing.

Sniffing sendiri dapat dikategorikan menjadi 2, yaitu aktif dan pasif. Sniffing pasif merupakan proses analisa paket jaringan tanpa melakukan perubahan atun pembuatan paket tertentu yang kemudian dikirimkan melelui jaringan. Sebaliknya sniffing aktif merupakan proses sniffing yang pada kondisi tertentu dapat melakukan perubahan ataupun pembuatan paket yang kemudian dikirimkan melalui jaringan.

* 1. Promiscuous Mode

*Promiscuous mode* atau *promisc mode* merupakan konfigurasi pada *Network Interface Card* (NIC) yang dapat menghambat atau meneruskan setiap paket yang melewatinya. Ketika NIC berada pada *promiscuous mode* maka setiap paket yang melewatinya (termasuk paket yang tidak ditujukan kepadanya) akan diteruskan ke CPU dan diproses.

* 1. Maximum Transmission Unit (MTU)

*Maximum Transmission Unit* (MTU) dalam jaringan komputer merupakan maksimum dari ukuran paket yang dapat ditransimikan oleh media jaringan. Ukuran dari MTU berfariasi tergantung pada media transmisi yang digunakan. Salah satu media transmisi yang umum digunakan adalah Ethernet dengan maksimum MTU adalah 1500 yang berarti paket yang ditransmisikan pada Ethernet Frame (*datalink layer*) tidak dapat melebihi 1500 *bytes.*

* 1. Pcap File Format

Format .pcap (Packet Capture) merupakan format standar yang digunakan untuk penyimpanan hasil *capture* data jaringan. Paket yang tersimpan di dalam format pcap tidak selalu berisi semua data seperti paket yang terdapat di jaringan jika *snaphot length* yang digunakan lebih kecil dari panjang paket yang terdapat di jaringan. Untuk mengatasi permasalahan ini kita dapat menerapkan *snapshot length* sepanjang 65535 (maksimum). Versi setelah pcap adalah pcap-ng, untuk lebih detailnya dapat dilihat di https://github.com/the-tcpdump-group/pcapng.

* 1. Jaringan Saraf Tiruan
  2. Backpropagation

**BAB III**

**METODE PENELITIAN**

1. **Obyek Penelitian**

*ARP* merupakan suatu protokol yang bertanggung jawab untuk mencari *Media Access Control (MAC) Address* dari setiap komputer yang akan berkomunikasi melalui jaringan *Local Area Network (LAN)* dengan memanfaatkan *Internet Protocol Address (IP Address)* versi 4 yang telah didapat saat sebuah komputer terkoneksi ke dalam jaringan. *MAC Address* yang telah didapat akan disimpan di dalam *ARP cache* agar tidak membuat jaringan sibuk dengan selalu mengirimkan paket *ARP* setiap kali ingin berkomunikasi dengan komputer lainnya.

Ketika sebuah host menerima *ARP* reply maka secara langsung akan meng-update *ARP cache* yang dimilikinya. Hal ini menyebabkan permasalahan ketika host tesebut mendapatkan *ARP reply* yang telah di buat sedemikian rupa oleh attacker (penyerang) karena pada proses update *ARP cache* tidak dilakukan pengecekan terlebih dahulu. Disini timbul permasalahan kerena *ARP cache* tersebut akan digunakan untuk berkomunikasi antar host yang terkoneksi di jaringan yang akan berakibat tidak sampainya data kepada tujuan yang diinginkan.

1. **Metode Penelitian**

**Pengumpulan Data**

Metode dan prosedur yang penulis gunakan untuk mendapatkan suatu data atau informasi tentang apa saja yang harus dikerjakan pada saat pengembangan sistem “Pendeteksi Serangan Pada ARP Menggunakan Jaringan Syaraf Tiruan” adalah sebagai berikut:

1. **Observasi**

Kegiatan yang dilakukan adalah dengan mengamati dan menganalisa setiap paket *ARP* yang dapat di-*capture* pada jaringan. Hasil dari kegiatan ini akan dijadikan acuan untuk menentukan metode yang tepat untuk menyelesaikan masalah.

1. **Analisis Kebutuhan**

Pada kegiatan ini akan dilakukan analisis kebutuhan sistem baik perangkat keras maupun perangkat lunak. Selain itu juga akan dilakukan analisis akan kebutuhan calon pengguna sistem yang dibuat akan tepat guna.

**Analisis Perancangan**

Dalam memenuhi kebutuhan pengguna, sistem ini membutuhkan dukungan hardware dan software diantaranya *Network Interface Card (NIC)*, Libpcap untuk GNU/Linux sebagai *packet capture library* dan Winpcap yang merupakan versi lain dari Libpcap untuk Windows dimana juga terdapat *network driver* untuk *packet capture*. Libpcap/Winpcap ini juga digunakan untuk menyimpan hasil dari paket-packet yang berhasil di-*capture*.Untuk penyimpanan bobot Jarigan Saraf Tiruan dan konfigurasi dari sistem digunakan format JSON.

**Pembuatan Program**

Sistem ini akan diimplementasikan dengan menggunakan bahasa pemrograman Java dan C (untuk pembuatam *library/modul* pengiriman dan pengambilan paket). Sedangkan penyimpanan hasil *capture* paket akan menggunakan format pcap ataupun pcapng.

**Implementasi dan Pengujian**

Sistem ini akan diimplementasikan pada beberapa komputer yang menggunakan sistem operasi Linux dan Windows, selain itu akan dilakukan beberapa kali pengujian sebelum dan saat sistem digunakan oleh pengguna.

**BAB IV**

**ANALISA DAN PERANCANGAN SISTEM**

1. **Analisis Sistem yang Berjalan**

Sistem ini melakukan deteksi dengan cara melakukan *filtering* terhadap paket *ARP* yang ditangkap oleh NIC (Network Interface Card) pada *promiscuous mode.* Setelah paket ARP diterima maka akan dilakukan ekstraksi (normalisasi) paket agar dapat digunakan sebagai inputan pada Jaringan Saraf Tiruan.

Jika paket yang diterima merupakan pakct ARP Reply (Ethertype: 0x0806, dan ARP Opcode: 2) maka akan dilakukan pengecekan untuk memastikan bahwa paket tersebut adalah paket yang memang ditujuan kepada sistem yang sedang digunakan. Jika paket tersebut memang ditujuan pada sistem yang sedang berjalan maka akan dilakukan proses ekstraksi (normalisai), sebagai berikut:

if (ethSrcHwAddr != arpSrcHwAddr || ethDstHwAddr != arpDstHwAddr) then

INV\_PKT = 1

else

INV\_PKT = 0

endif

if (storedEntry == nextObtainedEntry) then

USHA = 0

else if (onlyIpMatches) then

USHA = 1

else

// addNewEntry

USHA = 0

endif

if (ethIsPadded) then

UPETH = 0

else

UPETH = 1

endif

if (validOUI) then

OUI = 0

else if (unknowOUI)

OUI = 0.5

else

OUI = 1

endif

EPOCH\_TIME = [ selisih ]

Kelima data yang didapat dari hasil ekstraksi (normalisasi) paket tersebut akan diproses menggunakan Jaringan Saraf Tiruan dan akan menghasilkan sebuah *output* pada kisaran 0 (nol) sampai dengan 1 (satu). Jika *output* lebih kecil dari 0.5 maka sistem akan menganggap paket tersebut tidak berbahaya. Sebaliknya jika *output* lebih besar atau sama dengan 0.5 maka sistem akan menganggap paket tersebut berbahaya dan akan melakukan pengiriman paket *ARP Request* untuk memperbaharui *ARP cache-*nya dan juga mengirimkan paket *ICMP Trap* (merupakan paket *ICMP Reply* dengan *destination-*nya adalah *IP Address* sistem itu sendiri) dengan tujuan untuk melakukan pengecekan apakah penyerang mengaktifkan *IP Routing/IP Forwarding*-nya. Dengan mengetahui apakah *IP Routing/IP Forwarding* setidaknya kita dapat menduga-duga apa tujuan dari penyerang, apakah hanya ingin mendapatkan *bandwith* lebih atau lebih dari itu.

1. **Analisa Kebutuhan**
   1. **Kebutuhan Fungsional**

Sistem ini dapat menjalankan beberapa fungsi-fungsi yang dapat digunakan untuk optimalisasi deteksi serangan pada ARP, diantaranya:

* Dapat melakukan deteksi serangan *ARP spoofing* atau *ARP posoning.*
* Dapat melakukan aksi pencegahan dari serangan *ARP spoofing* atau *ARP posoning.*
  1. **Kebutuhan Non Fungsional**

Dibutuhkan beberapa *hardware* maupun *software* agar sistem ini dapat berjalan, diantaranya:

* Kebutuhan Perangkat Keras
  + NIC (Network Interface Card)
* Kebutuhan Perangkat Lunak
  + Winpcap/Npcap/Win10Pcap untuk sistem operasi Windows.
  + Minimal JRE 1.8.

1. **Analisa Pengembangan Sistem**

**DAFTAR PUSTAKA**

Kaur, I., 2013, Detection and Prevention of ARP Cache Poisoning, Thesis, Computer Science and Engineeting Department, Thapar University, Patiala.

Srinath, D., Panimalar S., Simla, A. J., dan Deepa, J., 2015, Detection and Prevention of ARP Spoofing using Centralized Server, Internation Journal of Computer Applications*, 113,* Departement of Computer science and Engineering, Panimalar Institute of Technology, India.

Vinay, K.R., Gudur, B.K., 2014, ARP Spoof Detection System Using ICMP Protocol: An Active Approach, International Journal of Engineering Research and Technologi (IJERT), Vol 3.