Implementasi dan Deteksi Serangan *Man-In-The-Middle* Berbasis MITM Proxy Terhadap Protokol HTTPS Menggunakan Metode K-NN

Rifki Rizaldi Setiadi[1], VeraSuryani[2], MuhammadAgusTriawan[3]

[1,2,3] Universitas Telkom, Bandung
rifkirizaldi@students.telkomuniversity.ac.id[1], verasuryani@telkomuniversity.ac.id[2], matriawan@telkomuniversity.ac.id[3]

Abstrak

Website menjadi media yang perkembangannya sangat pesat. Banyak metode keamanan yang diterapkan untuk mengamankan data yang tersimpan pada sebuah *website*, namun dari sisi pengguna sering terjadi kecerobohan ketika mengakses *website* dengan protokol *Hypertext Transfer Protocol Secure* (HTTPS) menggunakan perangkat *smartphone*, kecerobohan ini dapat menjadi sebab bocornya informasi rahasia yang terdapat pada *website* dan menjadi celah bagi serangan yang dilakukan oleh *attacker*. Salah satu contoh serangan yang sering terjadi kepada pengguna saat mengakses *website* HTTPS adalah serangan *Man-in-the-Middle* (MITM). Serangan MITM bekerja sebagai *broker* antara pengguna dan *wifi* dengan kemanan yang terbuka, salah satu pengembangan dari serangan MITM adalah MITM proxy. MITM proxy mampu melihat lalu lintas jaringan serta membuat sertifikat palsu ketika pengguna mengakses sebuah *website*. Data sensitif pengguna bisa terlihat dan didapatkan ketika serangan dilakukan, namun serangan yang terjadi menimbulkan anomali pada nilai *Round Trip Time* dan *Throghput*. Dalam mendeteksi serangan dan anomali yang terjadi, algoritma K-*Nearest Neighbor* (K-NN) berjalan dengan baik dan dapat digunakan untuk mendeteksi. Diharapkan hasil dari penelitian ini memberikan gambaran mengenai pendeteksian dan hasil serangan MITM proxy. Dari hasil serangan yang dilakukan, informasi sensitif pengguna yaitu *username* dan *password* berhasil didapatkan serta hasil pengujian deteksi serangan menggunakan algoritma K-NN memiliki nilai akurasi sebesar 95.1% dengan *error rate* sebesar 4.9%.

Kata kunci : ***Keamanan, Man-in-The-Middle, Algoritma K-Nearest Neighbor, Deteksi Serangan.***

Abstract

Website is one of the fastest-growing media. Many security methods are applied to secure data stored on a website, but from the user's perspective, carelessness can occur when accessing websites with the Hypertext Transfer Protocol Secure (HTTPS) protocol using a smartphone device. This carelessness can be the reason that all sensitive data can be accessed and provide an attack from attacker. One example of an attack that often occurs to users when accessing HTTPS websites is Man in the Middle (MITM) attack. MITM attacks serve as an intermediary between users and wifi with open security, one of the developments of MITM attacks is the MITM proxy. MITM proxies are capable of viewing network traffic as well as generating fake certificates when users access websites. Sensitive user data can be seen and obtained when the attack is carried out, but the attack that occurs causes anomalies in the Round Trip Time and Throughput values. In detecting attacks and anomalies that occur, the K-Nearest Neighbor (K-NN) algorithm can be used. It is hoped that the results from study can provide an overview about detection and MITM proxy attacks. From the results of the attacks, user-sensitive information such as usernames and passwords, are successfully obtained, and the results of attack detection tests using the K-NN algorithm had an accuracy value of 95.1% with an error rate of 4.9%.

Keywords: ***Security, Man-in-The-Middle, K-Nearest Neighbor Algorithm, Attack Detection.***

1. Pendahuluan

Latar Belakang

Kemanan ketika membangun *website* menjadi sebuah faktor penting yang sangat diperhatikan karena menyangkut kerahasiaan, data pribadi, dan celah yang memungkinkan dapat merusak *website* di kemudian hari. Dalam keamanan jaringan, terdapat *Transport Layer Security* atau disebut *Secure Sockets Layer*. Pada layer ini terdapat protokol komunikasi yang dienkripsi yaitu *Hypertext Transfer Protocol* (HTTP) yang dikembangkan menjadi *Hypertext Transfer Protocol Secure* (HTTPS). Dalam arsitektur jaringan *Transmission Control Protocol / Internet Protocol* (TCP/IP), HTTP terletak pada layer aplikasi dimana pada layer ini berfungsi untuk mendefinisikan dan menjabarkan aplikasi apa saja yang berjalan di dalam jaringan.

Hampir seluruh aplikasi *website* menggunakan protokol HTTPS dalam keamanan transaksi hingga data sensitif yang tersimpan pada *website*. Protokol HTTPS dipercaya untuk mencegah tampilan yang tidak sah dan informasi

rahasia pada *website* [1]. Dengan semakin banyaknya ancaman keamanan di dunia maya, maka faktor-faktor lain selain penerapan protokol HTTPS dalam sebuah *website* perlu diperhatikan. Kesalahan dalam konfigurasi dapat berpotensi didapatkannya kunci dari mesin *server* yang bersangkutan dan membaca seluruh data yang dikirim atau diterima oleh *server* ke pengguna tanpa adanya pengamanan sama sekali [2]. Kesalahan dari sisi pengguna merupakan faktor yang berakibat bocornya data sensitif sebuah *website* atau data sensitif pengguna itu sendiri. Pengguna yang mengakses internet melalui *public wifi* bisa menjadi sasaran empuk bagi attacker dalam melancarkan serangan untuk mencuri data, sebagian besar pengguna tidak peduli dan mengerti jika terdapat kendala dalam mengakses website, salah satu contoh kendala yang sering dijumpai adalah peringatan mengenai koneksi yang tidak aman ketika mengakses *website*. Dalam penelitian sebelumnya [3][4], disebutkan bahwa *Man-in-the-Middle* (MITM) Attack adalah serangan yang paling sering terjadi terhadap perangkat pengguna, aplikasi web, dan *website*.

Dari kasus yang dibahas pada paragraf kedua, hal tersebut menjadi perhatian karena serangan MITM mempunyai berbagai macam metode dan cara dalam mengimplementasikannya. Setiap metode serangan MITM memiliki karakteristik tersendiri terhadap perangkat yang diserang. Dalam penelitian [1][11] disebutkan serangan MITM yang dilakukan pada sistem *smart lock* memiliki peningkatan *Round Trip Time* (RTT) dan penurunan nilai *throughput*. Berdasarkan tantangan yang sudah dijelaskan dan penelitian yang sebelumnya dilakukan, penulis menggunakan implementasi serangan MITM dengan MITM proxy pada perangkat yang berbeda yaitu *smartphone* serta menggunakan K-Nearest Neighbor untuk mendeteksi dan mengukur keakuratan serangan yang terjadi.

Topik dan Batasannya

Topik dalam penelitian ini yaitu bagaimana mengimplementasikan serangan MITM proxy ketika pengguna mengakses *website* HTTPS melalui *public Wifi*, mendapatkan data sensitif pengguna yang diserang, dan mendeteksi anomali menggunakan metode klasifikasi K-NN dari sisi pengguna.

Adapun batasan masalah dalam penelitian ini yaitu pertama, penulis hanya mengimplentasikan serangan pada lingkungan *public Wifi* dengan keamanan yang terbuka. Kedua, deteksi anomali menggunakan *dataset* yang terdiri dari RTT dan *throughput*. Ketiga, data sensitif yang didapatkan berupa *username* dan *password* pengguna yang diserang.

Tujuan

Tujuan dari penelitian ini adalah mengimplementasikan serangan MITM proxy saat pengguna mengakses *website* HTTPS untuk mendapatkan data berupa *username* dan *password* serta mendeteksi anomali berdasarkan *dataset* RTT dan *throughput* ketika serangan terjadi menggunakan metode K-NN dari sudut pandang pengguna yang diserang.

2. Studi Terkait

Peneliti Fajrin,Sukarno, dan Satwiko [1] melakukan penelitian berjudul "Perbandingan Metode K-NN dan Markov Chain Untuk Deteksi Anomali Serangan Man in the Middle Pada Smart Lock Berbasis Wifi" pada tahun 2020. Peneliti melakukan penelitian metode klasifikasi K-NN dan Markov Chain dengan menggunakan *dataset* berupa hasil serangan MITM terhadap *smart lock*. Pada penelitian ini membandingkan kedua algoritma metode tersebut dengan tujuan untuk menentukan algoritma yang memiliki pendeteksian serangan dengan keakuratan yang tinggi dan perbandingan disesuaikan dengan evaluasi algoritma. Dari hasil pengujian didapatkan bahwa algoritma K-NN lebih baik dari algoritma Markov Chain, dimana akurasi algoritma K-NN yaitu 95,6 persen dibandingkan dengan Markov Chain yaitu 51,4 persen. Namun dalam penelitian tersebut tidak dijabarkan secara rinci proses penyerangan yang dilakukan terhadap *smart lock* berbasis *Wifi*. Maka penelitian ini menjelaskan bagaimana serangan MITM dilakukan menggunakan MITM proxy dan perangkat yang diserang berbeda dengan penelitian tersebut.

Peneliti Wahannani, Aditiawan, dan Mimpuni [3] melakukan penelitian berjudul "Uji Coba Serangan Man-in-the-Middle Pada Keamanan SSL Protokol HTTP" pada tahun 2020. Peneliti menguji serangan terhadap keamana SSL dengan HTTP sebagai protokol keamanan menggunakan beberapa *tools sniffing*. Hasil yang didapatkan, keamanan SSL dengan HTTP sebagai protokolnya sangat bagus, terbukti ketika aktifitas *login* direkam. Paket data dapat dienkripsi dengan baik sehingga *username* dan *password* sulit untuk didapatkan. Pada penelitian tersebut diusulkan untuk menggunakan metode MITM yang lain selain teknik *sniffing*. Maka penelitian ini akan menggunakan metode MITM dengan ARP *spoofing* menggunakan MITM proxy.

Peneliti Anand, Prathiba, Gunasekaran, dan Ponmani [4] melakukan penelitian berjudul " Detection of Man-in-the-Middle Attacks in WiFi Networks by IP Spoofing" pada tahun 2018. Peneliti melakukan penelitian mengenai algoritma *MannWhiteyU* yang dapat mendeteksi adanya serangan MITM berbasis IP *Spoofing*. Dalam hasil algoritma yang telah dibuat, peneliti mendapati bahwa dengan lonjakan rentang waktu target yang lebih sering, ditemukan ada serangan MITM yang diterapkan pada mesin target di jaringan *wifi*. Menganalisis titik data dengan uji *MannWhitneyU* dapat memisahkan lonjakan abnormal dari titik data normal. Pada penelitian tersebut

serangan yang dilakukan hanya ditujukan untuk mendapatkan anomali, tidak dijelaskan apakan metode IP *spoofing* dapat melihat data pengguna ketika serangan dilakukan. Maka dari itu penelitian ini menjelaskan dan menguji dengan metode ARP spoofing menggunakan MITM proxy untuk melihat data pengguna ketika serangan terjadi.

Peneliti Setiyadi [5] melakukan penelitian berjudul "Implementasi Modul Network MITM Pada Websploit sebagai Monitoring Aktifitas Pengguna dalam Mengakses Internet" pada tahun 2017. Peneliti melakukan penelitian mengenai MITM pada *websploit* guna memonitor aktifitas pengguna ketika menjelajah internet. Dalam implementasi yang dilakukan, peneliti menentukan *website* unikom dan kompas sebagai target. Hasil yang didapatkan oleh peneliti yaitu berupa informasi perangkat pengguna yang sukses didapatkan ketika mengakses kedua halaman *website* tersebut dan dapat disimpulkan bahwa MITM pada *websploit* dapat memonitor aktifitas pengguna. Pada penelitian tersebut belum dijelaskan mengenai anomali pada RTT dan *throughput* ketika serangan dilakukan. Maka dari itu penelitian ini menguji anomali serangan MITM menggunakan klasifikasi K-NN.

Peneliti Kamajaya, Riadi, dan Prayudi [6] melakukan penelitian berjudul "Analisa Investigasi Static Forensics Serangan Man-in-the-Middle Berbasis ARP *Poisoning*" pada tahun 2020. Peneliti melakukan penelitian berupa analisa secara langsung terhadap serangan MITM dengan teknik ARP *poisoning*. Pendekatan dengan metode statik forensik dilakukan guna mendeteksi aktivitas yang ilegal pada jaringan *Wifi*. berfokus pada analisa network trafik, proses untuk menemukan barang bukti dari serangan MITM dengan teknik ARP *Poisoning*. Hasil dari penelitian ini yaitu menganalisa dan menemukan bukti maupun informasi pelaku. Dalam penelitian tersebut untuk mendeteksi serangan masih dilakukan secara manual dengan melihat *data traffic* ketika serangan terjadi. Maka dari itu penelitian ini menggunakan metode klasifikasi K-NN dalam mendeteksi serangan MITM agar memudahkan analisa dari sisi pengguna yang diserang.

Peneliti Pingle, Mairaj, dan Javaid [8] melakukan penelitian berjudul "Real World Man in the Middle (MITM) Attack Implementation Using Open Source Tools for Instructional Use" pada tahun 2018. Peneliti melakukan penelitian berupa implementasi serangan *Man-in-the-Middle* dengan berbagai macam *Open Source Tools*. Dalam penelitian ini, peneliti menjelaskan segala bentuk serangan *Man-in-the-Middle* dan memberikan instruksi dalam menggunakan *tools* yang akan di implementasikan. Ruang lingkup yang digunakan oleh peneliti adalah *Virtual Box*, dimana serangan yang dilakukan menggunakan *Ettercap Graphical*. Pada penelitian tersebut ruang lingkup yang digunakan adalah virtual box. Maka dari itu penelitian ini menggunakan ruang lingkup secara *live* terhadap perangkat yang asli dalam mengimplementasikan serangan MITM.

2.1 K-Nearest Neighbor

K-Nearest Neighbor (K-NN) merupakan metode klasifikasi objek berdasarkan atribut dan *training sample*. Sebelum menjalankan perhitungan dengan metode K-NN, perlu ditentukan data latih dan data uji. Lalu dilakukan perhitungan untuk mencari jarak dengan formula *Euclidean distance*.

$$\text{d}(\text{a},\text{b}) = \sqrt{\sum_{\text{i}=1}^{\text{n}} (\text{a}_{\text{i}} + \text{b}_{\text{i}})^2} \quad (1)$$

Dimana d(a,b) merepresentasikan jarak (*distance*) dari a ke b dan n merepresentasikan jumlah parameter yang ada. Dalam pengklasifikasian K-NN seluruh data latih dilihat untuk mencari poin-poin/jarak terdekat dan diberi label K. Nilai K yang dipilih berjumlah ganjil untuk menghindari jumlah klasifikasi yang kembar. klasifikasi yang kembar terjadi karena algoritma K-NN mengandalkan *major vote* dimana mayoritas yang terbanyak akan dipilih sebagai label yang dicari, jika label berjumlah genap maka akan timbul kemungkinan jumlah seri [1]. Setelah itu, dengan algoritma K-NN akan dilakukan tahapan perhitungan klasifikasi objek berdasarkan data yang jaraknya paling dekat dengan objek satu dan objek yang lainnya.

2.2 Evaluasi Algoritma K-NN

Untuk mengevaluasi performansi klasifikasi K-NN pada sistem yang dibangun dapat menerapkan metode *confusion matrix*, metode ini berisikan informasi klasifikasi yang aktual dan terprediksi menggunakan unsur-unsur yang ditunjukan pada tabel 1.

Tabel 1 Unsur-Unsur *Confusion Matrix*

		Predicted	
		Positive	Negative
Actual	Positive	True Positive	False Negative
	Negative	False Positive	True Negative

True Positive (TP) adalah label positif yang diprediksi benar, *False Negative* (FN) adalah label positif yang diberi nilai negatif, *False Positive* (FP) adalah label negatif yang diberi nilai positif, lalu *True Negative* (TN) yang

merupakan label negatif yang diprediksi dengan benar. Pada *confusion matrix* terdapat beberapa parameter yang digunakan untuk melihat performansi yang dituangkan dalam persamaan berikut.

$$AC = \frac{TP + TN}{TP+TN+FP+FN} \quad (1)$$

dimana *accuracy* (AC) merupakan parameter evaluasi dalam mengukur keakuratan pada sistem klasifikasi yang dibangun.

$$PC = \frac{TP}{TP+FP} \quad (2)$$

dimana *precision* (PC) merupakan jumlah label positif yang diprediksi benar lalu dibagi dengan keseluruhan label positif.

$$RC = \frac{TP}{TP+FN} \quad (3)$$

dimana *recall* (RC) adalah rasio total dari jumlah klasifikasi yang positif lalu dibagi dengan keseluruhan jumlah yang positif.

$$F1-Score = \frac{2\,(Precision \times Recall)}{Precision+Recall} \quad (4)$$

dimana F1-*Score* merupakan nilai rata-rata *precision* dan *recall* guna didapatkannya nilai *precision* dan *recall* yang seimbang.

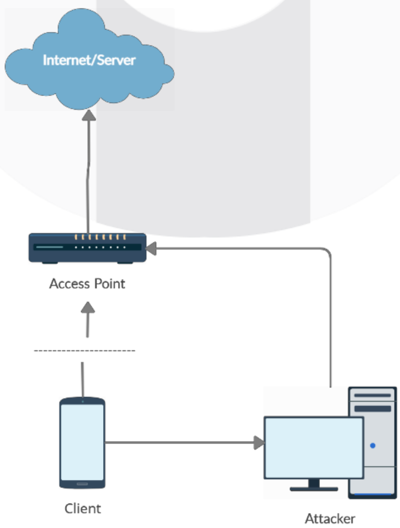
$$Error\ rate = \frac{FP + FN}{TP+FP+TN+FN} \quad (5)$$

dimana *error rate* merupakan jumlah seluruh label yang diprediksi salah lalu dibagi dengan keseluruhan label yang telah diprediksi.

3. Sistem yang Dibangun

3.1 Topologi dan Kebutuhan Sistem

Topologi dan kebutuhan untuk serangan ini terdiri dari tiga perangkat yaitu *client*, *router Wifi* sebagai *Access Point*, dan perangkat *attacker* sebagai pelaku yang menjalakan serangan.



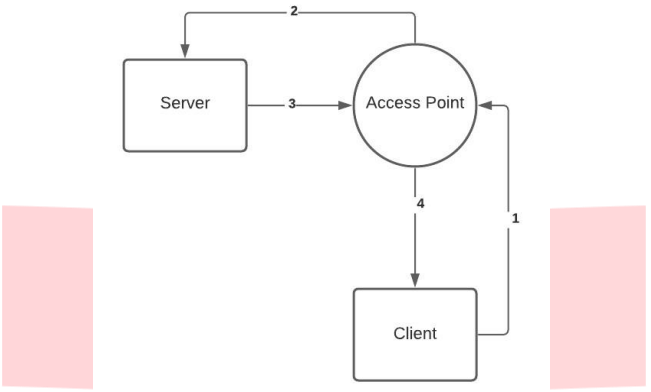
Gambar 1 Topologi Serangan

Dalam topologi serangan yang terdapat pada gambar 1, pengguna selaku *client* menggunakan perangkat SM-G610F dengan sistem android 8.1.0 menggunakan *security software version* SMR Apr-2020, pelaku serangan

selaku *attacker* menggunakan perangkat laptop dengan sistem kali linux versi 2021.1 64-bit, dan *Wifi* selaku *access point* menggunakan perangkat *router* Tp-Link dengan keamanan yang terbuka.

3.2 Skema Umum

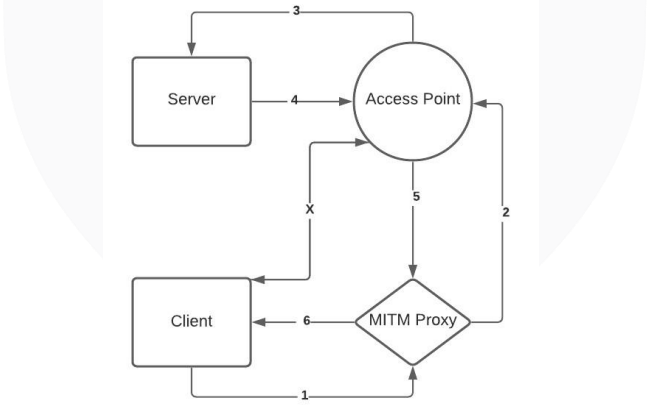
Dalam skema umum, *client* mengakses *website* HTTPS melalui *access point* yang menghubungkan *client* dengan *server*. *Client* akan meminta *public key* untuk mengakses *website* HTTPS dan *website* akan mengirimkan sertifikat *website* tersebut. Setelah mendapatkan sertifikat maka *client* akan menggunakan sertifikat tersebut untuk divalidasi oleh *Certificate Authority* (CA). Setelah sertifikat tervalidasi maka *client* dan *server* akan saling mengirimkan *private key* yang di enkripsi untuk pengiriman data satu sama lain. Jika sertifikat yang dipakai *client* tidak valid, maka koneksi tidak akan pernah terhubung, *client* dan *server* akan terhubung ketika sertifikat berhasil di validasi dan saling mengirimkan *private key*.



Gambar 2 Aliran Data Pada Sistem

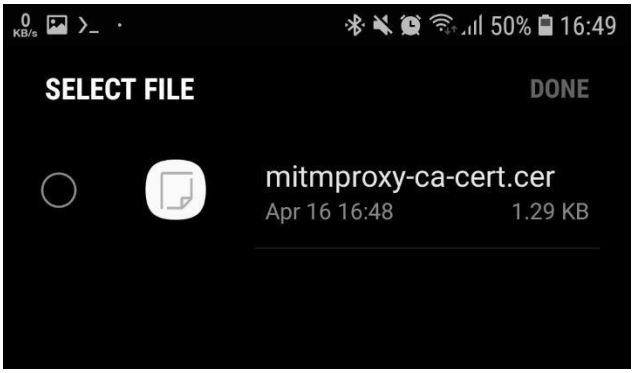
Pada Aliran data yang terjadi di gambar 2, ketika *client* ingin terhubung dengan *server* maka *client* perlu mengakses *access point* (1) untuk mendapatkan akses internet, setelah akses didapatkan maka *access point* akan meneruskan permintaan *client* agar dapat terhubung kepada *server* (2). Ketika *server* berhasil dihubungi, maka *server* akan merespon panggilan tersebut dengan mengirimkan pesan kepada *access point* (3). Setelah *access point* menerima pesan balasan, maka *access point* akan meneruskan kembali jawaban tersebut kepada *client* (4). Proses tersebut akan terus berulang selama *client* mengakses *server* dan seluruh aliran data akan melalui jalur-jalur tersebut.

3.3 Serangan MITM Proxy Pada Sistem



Gambar 3 Aliran Data Pada Sistem Ketika Diserang

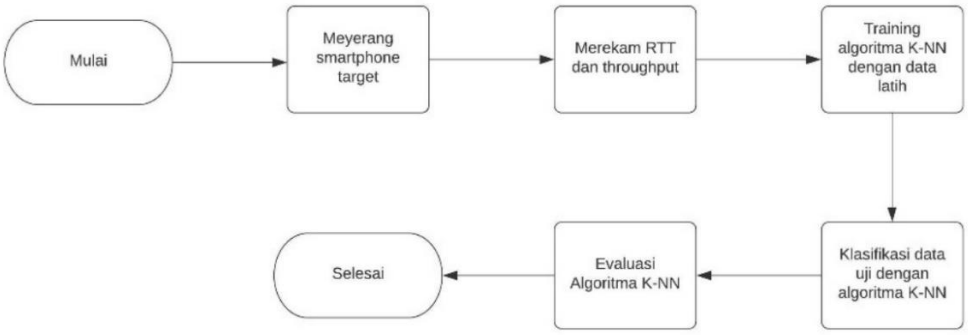
Pada gambar 3, serangan MITM proxy yang terjadi bertindak sebagai *broker* antara *client* dan *access point*. Dimana aliran data yang *client* kirimkan (1) akan melalui MITM proxy dan meneruskannya menuju *access point* (2) ketika proses handshake terjadi maka MITM proxy akan mengirimkan setifikat palsu sehingga *client* dapat mengakses *website* HTTPS. Aliran data yang terjadi (1-2-3) dan (4-5-6) tidak akan terputus selama *client* mengizinkan sertifikat palsu untuk mengakses *website* HTTPS. Ketika serangan dilakukan, terjadi anomali dengan meningkatnya RTT dan menurunnya nilai *throughput* [1][11] pada sisi *client*. Jika MITM proxy gagal untuk memberikat sertifikat palsu dan sertifikat palsu tersebut tidak divalidasi, maka akan ada peringatan yang dikeluarkan oleh peramban mengenai koneksi jaringan tidak aman untuk mengakses *website* HTTPS seperti pada gambar 8 dan gambar 9.



Gambar 4 *File* Sertifikat Palsu

Jika sertifikat yang diduplikasi oleh MITM proxy gagal untuk validasi, maka diperlukan cara manual untuk memasang sertifikat kedalam *smartphone* pengguna. Terdapat berbagai macam cara dalam memasukan *file* dan meninstal sertifikat palsu kedalam *smartphone* pengguna, Bergantung pada kreatifitas *attacker*, cara yang dapat dilakukan adalah *social engineering* atau teknik rekayasa lainnya untuk menjebak pengguna agar sertifikat bisa terpasang. Namun dalam penelitian ini tidak dibahas mengenai pemasangan sertifikat secara manual dan pada *smartphone* pengguna sudah terpasang sertifikat palsu tersebut. Pada gambar 4 menunjukan sertifikat palsu berupa *file* yang sudah dimasukan ke dalam *smartphone* pengguna dan sudah terpasang.

3.4 Skema Pengujian



Gambar 5 Flowchat Skema Pengujian

Pada gambar 5, diperlihatkan skema pengujian dalam melakukan penelitian dengan urutan menyerang *smartphone* target, merekam RTT dan *throughput*, *training* algoritma K-NN dengan data latih, klasifikasi data uji dengan algoritma K-NN, dan evaluasi algoritma K-NN. Dalam tahapan klasifikasi data uji dengan algoritma K-NN, data uji diberi label positif dan negatif berdasarkan *training* algoritma K-NN dengan data latih. Tahap evaluasi algoritma digunakan untuk mencari tingkat akurasi dan error ratenya lalu ditarik kesimpulan apakah metode klasifikasi K-NN dapat mendeteksi serangan yang terjadi pada perangkat *smartphone* berdasarkan nilai akurasi yang didapatkan.

3.5 Persiapan Implementasi MITM Proxy

Pada tahap persiapan untuk melakukan implementasi, dibutuhkan perangkat penyerang dan perangkat pengguna yang sudah terhubung pada *public wifi*. Tahapan dalam melakukan implementasi serangan dilakukan dengan cara berikut.

1. Konfigurasi

   Dalam tahap ini, penyerang melakukan konfigurasi dengan menjalankan perintah yang dilakukan pada teminal kali linux, hal yang diperhatikan dalam melakukan konfigurasi yaitu

   a. Merubah ip_forward

      Merubah pengaturan ip_forward dilakukan untuk merubah status parameter *kernel*. Untuk merubah ip_forward perlu memasukan perintah pada terminal sebagai berikut.

```
# echo 1 > /proc/sys/net/ipv4/ip_forward
```

echo 1 berfungsi untuk mengaktifkan *forwarding* sedangkan untuk mematikan *forwarding* yaitu *echo 0*.

```
# sysctl -w net.ipv4.ip_forward = 1
```

Sama halnya seperti perintah sebelumnya, forward=1 untuk mengaktifkan *forwarding* untuk protokol ipv4.

b. Mengalihkan tcp *port* 80 dan 443 ke *port* 8080

Tcp *port* 80 dan 443 perlu dialihkan karena MITM proxy bekerja pada *port* 8080, mengalihkan *port* tcp tersebut agar lalu lintas jaringan terlihat dan dapat mendapatkan data yang dibutuhkan. Untuk mengalihkan *port* 80 dan 443 perlu memasukan perintah pada terminal sebagai berikut.

```
# iptables -t nat -A PREROUTING -i wlan0 -p tcp --dport 80 -j REDIRECT --to-port
8080
```

Perintah tersebut digunakan untuk mengalihkan *port* 80 menjadi *port* 8080 pada perangkat *Wireless Local Area Network* (WLAN),

```
# iptables -t nat -A PREROUTING -i wlan0 -p tcp --dport 443 -j REDIRECT --to-port
8080
```

sama halnya seperti perintah pada *port* 80, perintah diatas untuk mengalihkan *port* 443 menjadi *port* 8080 pada perangkat WLAN.

2. Melakukan ARP *Spoofing*

Dalam melakukan ARP *Spoofing*, *route gateaway* dan *network interface* perangkat penyerang perlu diketahui telebih dahulu, Sepeti yang terlihat pada gambar 10. Untuk mengetahui *interface* perangkat penyerang dapat memasukan perintah pada terminal sebagai berikut.

```
# ifconfig
```

Setelah *network interface* dan IP *Address* penyerang diketahui, maka langkah selanjutnya adalah mencari *network gateaway* pada *Wifi* yang terbuhung pada perangkat penyerang, seperti yang terlihat pada gambar 11. Untuk mengetahui *route gateaway wifi* perangkat penyerang dapat memasukan perintah pada terminal sebagai berikut.

```
# route -n
```

Untuk mencari target IP *Address*, Nmap harus dijalankan terlebih dahulu untuk memindai seluruh IP *Address* yang terdaftar dan terhubung dengan *Wifi* tersebut, seperti yang terlihat pada gambar 12. Untuk mengetahui perangkat apa saja serta *port* mana saja yang terbuka dan terhubung ke *router Wifi* dapat memasukan perintah pada terminal sebagai berikut.

```
# nmap (IP Gateaway)
```

Setelah data didapatkan, maka dapat melakukan ARP *Spoofing* kepada *smartphone* pengguna dan *public wifi* seperti yang terlihat pada gambar 13. Untuk melakukan ARP *spoofing* dapat memasukan perintah pada terminal sebagai berikut.

```
# arpspoof -I wlan1 -t IProuter IPclient
```

Perintah tersebut digunakan pada kedua IP *Address*, mengalihkan IP *router wifi* dan taget IP *address* ke perangkat penyerang.

3.6 Implementasi Serangan MITM Proxy

Skenario implementasi serangan MITM proxy sesuai dengan rancangan sistem. Aktifitas pengguna berupa *login* kedalam *website* HTTPS, *Website* yang diakses yaitu www.twitter.com dan www.igracias.telkomuniversity.ac.id. Diasumsikan bahwa *smartphone* pengguna telah terpasang sertifikat palsu yang diunduh oleh *attacker* dari *website* mitmproxy. Sertifikat palsu yang telah dipasang pada *smartphone* pengguna akan tervalidasi dan memberikan akses kepada pengguna ketika mengunjungi *website* HTTPS. Untuk menjalankan dan mengakses MITM proxy yaitu dengan memasukan perintah kedalam terminal sebagai berikut.

```
# mitmproxy –mode transparent –ssl-insecure
```

Urutan perintah dalam line tersebut adalah “mitmproxy” untuk mengakses mitmproxy, “—mode” untuk menentukan mode *capture traffic* yang ingin didapatkan, “—ssl-insecure” untuk validasi sertifikat *web server*. Seperti yang terlihat pada gambar 14. Setelah perintah mitmproxy berhasil dijalankan maka lalu lintas jaringan antara *smartphone* pengguna dan *website* HTTPS didapatkan seperti yang tertera pada gambar 15.

3.7 Dataset

Dalam deteksi serangan dan evaluasi algoritma K-NN, data latih dan data uji diperlukan. Data latih didapatkan melalui simulasi serangan terhadap sistem yang mengakses *website* HTTPS, data latih dimodifikasi dengan diberi label positif dan negatif berdasarkan perbedaan nilai RTT dan *throughput*. Untuk data uji didapatkan melalui serangan *live* pada sistem yang sedang mengakses *website* HTTPS dan belum diberi label positif dan negatif. Kedua *dataset* memiliki jumlah data yang sama dan parameter yang serupa yaitu RTT dan *throughput*.

3.8 Deteksi Serangan MITM Proxy

Dalam mendeteksi serangan MITM proxy, data latih yang sudah diberi label positif dan negatif diolah dengan algoritma K-NN untuk mendapatkan nilai AC, PC, RC, dan F1-*score*. Nilai AC, PC, RC, dan F1-*score* dijumlahkan berdasarkan nilai k yang berbeda, nilai k yang diuji memiliki *range* nilai dari 1 hingga 40. Nilai k dengan jumlah AC, PC, RC, dan F1-*score* terbesar digunakan sebagai k terbaik. Setelah k terbaik didapatkan, data uji diolah untuk diberi label positif dan negatif berdasarkan prediksi algoritma K-NN terhadap data latih.

4. Evaluasi dan Hasil

Dalam penelitian ini, terdapat tiga skenario yang akan dilakukan. Skenario yang dilakukan adalah implementasi serangan MITM proxy terhadap sistem yang mengakses *website* HTTPS, pendeteksian serangan MITM proxy menggunakan metode K-NN, dan evaluasi algoritma K-NN terhadap deteksi serangan MITM proxy.

4.1 Hasil Implementasi Serangan

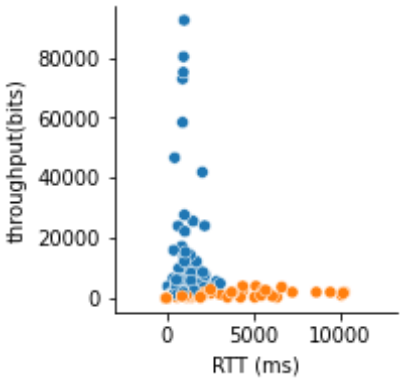
Hasil implementasi serangan MITM proxy pada *smartphone* pengguna yang mengakses *website* HTTPS berhasil dan dapat dilakukan. Hasil yang didapatkan adalah lalu lintas jaringan ketika pengguna menelusuri *website* seperti pada gambar 15. Informasi sensitif berupa *username* dan *password* yang menjadi tujuan pada penelitian ini berhasil didapatkan ketika pengguna *login* kedalam *website* seperti pada gambar 16 dan 17.

Tabel 2 Informasi Target Pengguna pada *website* HTTPS

	www.twitter.com	www.igracias.telkomuniversity.ac.id
Username	Rfkirs	rifkirizaldi
Password	7a******g	@*****d9***

Pada tabel 2 ditunjukan *username* dan *password* pengguna yang didapatkan oleh MITM proxy. Ketika pengguna mengakses kedalam *website* twitter, *username* yang terlihat adalah ‘Rfkirs’ dengan *password* ‘7a******g’. Untuk *website* igracias.telkomuniversity.ac.id *username* pengguna yang terlihat adalah ‘rifkirizaldi’ dengan password ‘@*****d9***’.

4.2 Hasil Deteksi Serangan MITM Proxy



Gambar 6 Diagram Plot Deteksi Algoritma K-NN

Pada gambar 6, ditunjukan hasil deteksi serangan MITM proxy dengan diagram plot dimana titik berwarna biru merepresentasikan bukan serangan dan titik berwarna oranye merepresentasikan serangan. Algoritma K-NN yang memanfaatkan *euclidean distance* pada sebuah titik yang menjadi subjek pencarian dapat mengelompokan anomali kedalam serangan atau bukan serangan. Anomali RTT dan throughput yang menjadi parameter pada serangan MITM proxy terjadi ketika terdapat perbedaan dan perubahan nilai pada kedua parameter tersebut. Lonjakan nilai RTT dan penurunan nilai *throughput* yang menyebabkan besarnya nilai RTT dari *throughput* menunjukan serangan pada sistem. Sebaliknya, penurunan nilai RTT dan lonjakan nilai *throughput* yang menyebabkan nilai RTT lebih kecil dari nilai *throughput* menunjukan bukan serangan pada sistem tersebut. Hasil yang didapatkan melalui *training* data latih pada algoritma K-NN terhadap data uji berjalan dengan baik, dataset hasil deteksi untuk data uji ditunjukan pada tabel 7. Namun terdapat kelemahan dalam mendeteksi menggunakan algoritma K-NN yaitu hasil deteksi ditentukan berdasarkan data latih, jika data latih memiliki data yang salah atau hasil prediksi yang kurang akurat, maka akan mempengaruhi deteksi algoritma K-NN terhadap *dataset* yang akan diuji setelahnya.

4.3 Hasil Evaluasi Algoritma K-NN

Hasil evaluasi algoritma K-NN terhadap data uji didapatkan melalui *training* data latih yang telah dilakukan sebelumnya. Nilai K yang digunakan adalah tiga, berdasarkan penjumlahan seluruh parameter AC, PC, RC, dan F1-*score* yang terbesar pada *training* data latih, nilai K terbesar ditunjukan pada tabel 3.

Tabel 3 Total Nilai Parameter Data Latih berdasarkan K

Nilai K	AC + PC + RC + F1-score
1	400
2	400
3	400
4	400
5	393.1306715
6	393.1306715
…	….

Hasil evaluasi algoritma K-NN terhadap data uji menunjukan nilai unsur-unsur *confusion matrix* dengan TP sebanyak 15 label, TN sebanyak 43 label, FP sebanyak 1 label, dan FN sebanyak 2 label yang ditunjukan pada tabel dibawah ini.

Tabel 4 Tabel *Confusion Matrix* Data Uji

	Predicted	
	Positive	Negative

Actual	Positive	15	2
	Negative	1	43

Berdasarkan tabel *confusion matrix* data uji, nilai AC dan error rate dapat diketahui dengan memasukan nilai TP,FP,TN, dan FN kedalam rumus AC dan error rate, maka didapatkan nilai AC sebesar 95.1% dengan *error rate* sebesar 4.9%. Hasil akurasi dan error rate yang diperoleh menunjukan tingkat keberhasilan algoritma K-NN dalam mendeteksi serangan yang terjadi. Pendeteksian dengan algoritma K-NN berhasil dan dapat digunakan untuk mendeteksi serangan karena tingkat akurasi lebih dari 90% dengan nilai error rate kurang dari 10%.

5. Kesimpulan dan Saran

Berdasarkan implementasi serangan MITM proxy pada *smartphone* pengguna yang mengakses *website* HTTPS, serangan dapat dilakukan serta *username* dan *password* pengguna didapatkan. Serangan MITM dengan teknik ARP *Spoofing* menggunakan MITM proxy dapat merekam seluruh komunikasi dan melihat informasi sensitif pengguna ketika mengakses *website*. Hasil deteksi menggunakan algoritma K-NN berjalan dengan baik dan berhasil mengelompokan jenis serangan atau bukan serangan dengan tingkat akurasi diatas 90% yaitu 95.1% dengan tingkat *error rate* dibawah 10% yaitu 4.9%. Dalam mencegah serangan MITM agar tidak terjadi yaitu selalu berhati-hati dalam mengakses sebuah *public Wifi* dengan keamanan yang terbuka, jangan menggunakan *Wifi* dengan nama yang mencurigakan, gunakan *Virtual Private Network* (VPN) ketika menggunakan *Wifi*, hindari perintah *log on* atau permintaan persetujuan apapun ketika menggunakan *public Wifi*. Saran untuk penelitian selanjutnya yaitu implementasi serangan dengan target dan perangkat yang berbeda serta pengembangan algoritma K-NN berbentuk aplikasi untuk mendeteksi serangan.

Referensi

[1] M. I. Fajrin, P. Sukarno, dan A. G. P. Satwiko, 2020, Perbandingan Metode K-NN dan Markov Chain Untuk Deteksi Anomali Serangan Man in the Middle Pada Smart Lock Berbasis Wifi, Fakultas Informatika, Universitas Telkom.

[2] W. S. Raharjo dan A. A. Bajuadji, 2016, Analisa Implementasi Protokol HTTPS pada Situs Web Perguruan Tinggi di Pulau Jawa, Program Studi Teknik Informatika Fakultas Teknologi Informasi UKDW Yogyakarta.

[3] H. E. Wahannani, F. P. Aditiawan, dan R. Mumpuni, 2020, Uji Coba Serangan Man-in-the-Middle Pada Keamanan SSL Protokol HTTP, Program Studi Informatika, Fakultas Ilmu Komputer, UPN"Veteran".

[4] G. Anand, S. B. Prathiba, Gunasekaran, dan Ponmani, 2018, Detection of Man-in-the-Middle Attacks in WiFi Networks by IP Spoofing, Department of Computer Technology, Madras Institute of Technology, Anna University.

[5] A. Setiyadi, 2017, Implementasi Modul Network MITM Pada Websploit sebagai Monitoring Aktifitas Pengguna dalam Mengakses Internet, Jurusan Teknik Informatika, FTIK UNIKOM.

[6] G. E. A. Kamajaya, I. Riadi, dan Y. Prayudi, 2020, Analisa Investigasi Static Forensics Serangan Man-in-the-Middle Berbasis ARP Poisoning, Program Studi Magister Teknik Informatika – Universitas Islam Indonesia dan Program Studi Sistem Informasi, Universitas Ahmad Dahlan.

[7] P. K. Pateriya dan S. S. Kumar, 2012, Analysis on Man-in-the-Middle Attack on SSL, Computer Science Department Lovely Professional University.

[8] B. Pingle, A. Mairaj, dan A. Y. Javaid, 2018, Real World Man in the Middle (MITM) Attack Implementation Using Open Source Tools for Instructional Use, Electrical Engineering and Computer Science Department, University of Toledo.

[9] Z. Chen, S. Guo, R. Duan, dan S. Wang, 2009, Security Analysis on Mutual Authentication agains Man in the Middle Attack, The Institute of North Electronic Equipment.

[10] J. J. Fritz, J. Sagisi, J. James, A. St. Leger, K. King, dan K. J. Duncan, Simulation of Man-in-the-Middle Attack on Smart Grid Testbed, Electrical Engineering and Computer Science, United States Military Academy.

[11] V. A. Valliavaara, M. Sailio, dan K. Halunen, 2014. Detecting Man in The Middle Attacks on Non Mobile System, Technical Research Center of Finland.