JITET (Jurnal Informatika dan Teknik Elektro Terapan)
Vol. 12 No. 1, pISSN: 2303-0577 eISSN: 2830-7062 http://dx.doi.org/10.23960/jitet.v12i1.3842

SISTEM DETEKSI DAN PENCEGAHAN INTRUSI PORTABEL TERHADAP SERANGAN DHCP STARVATION PADA JARINGAN NIRKABEL

Bahar[1*], Yuyun Wabula[2], Andani Ahmad[3]

[1,2,3]Universitas Handayani Makassar, Jl. Adhyaksa Baru No.1, Makassar, Sulawesi Selatan, Indonesia

Riwayat artikel:

Received: 13 Desember 2022
Accepted: 29 Desember 2023
Published: 1 Januari 2024

Keywords:
Denial of Service, DHCP Starvation, ARP Protocol Analysis, IDPS, Python.

Corespondent Email:
baharpalopo212@gmail.com

Abstrak. Layanan DHCP menjadi sangat penting karena merupakan penentu keberhasilan koneksi sekitar 22,18% dari 210 juta pengguna yang terkoneksi ke internet melalui jaringan nirkabel di Indonesia. Layanan DHCP tidak luput dari ancaman keamanan, salah satunya adalah serangan *Denial of Service* bernama DHCP starvation yang akan mengganggu ketersediaan layanan DHCP. Teknik baru yang digunakan pada serangan tersebut, yaitu dengan memanipulasi protokol ARP pada saat DHCP server melakukan deteksi konflik IP di jaringan. Peneliti mengusulkan menggunakan metode berbasis *Protocol Analysis* terhadap protokol ARP untuk melakukan deteksi dan pencegahan, maka dibuatlah sebuah piranti portabel yang berfungsi sebagai IDPS (*Intrusion Detection and Prevention System*) agar tidak hanya efektif namun juga lebih efisien. Aplikasi IDPS dibuat menggunakan bahasa Python dengan bantuan pustaka Scapy dan Paramiko. Hasil deteksi dan pencegahan menunjukkan tingkat akurasi sebesar 100%.

Abstract. *The DHCP service has become highly critical as it plays a pivotal role in the connectivity success of approximately 22.18% of 210 millions users who access the internet through wireless networks in Indonesia. DHCP services are not exempt from security threats, one of which is a Denial of Service attack known as DHCP starvation, capable of disrupting the availability of DHCP services. A novel technique employed in this attack involves manipulating the ARP protocol when the DHCP server detects IP conflicts within the network. Researchers propose an ARP protocol-based analysis method for detection and prevention, leading to the development of a portable device that serves as an Intrusion Detection and Prevention System (IDPS). This approach aims to not only enhance effectiveness but also improve efficiency. The IDPS application is crafted using the Python, with support from Scapy and Paramiko libraries. The results of detection and prevention show an accuracy rate of 100%.*

1. PENDAHULUAN

Berdasarkan survei yang dilakukan oleh APJII pada kuartal pertama tahun 2022, terhitung sekitar 210 juta pengguna internet di Indonesia, sebesar 22,18% diantaranya adalah pengguna internet yang menggunakan koneksi jaringan nirkabel [1]. Untuk memberikan kemudahan akses ke jaringan internet, maka layanan DHCP (*Dynamic Host Configuration Protocol*) digunakan untuk melakukan konfigurasi alamat IP (*Internet Protocol*) secara otomatis di sisi perangkat pengguna [2], itu sebabnya layanan DHCP digunakan secara luas dan memiliki peran yang sangat penting dalam sebuah jaringan nirkabel [3]. Namun ternyata layanan DHCP tidak luput dari ancaman keamanan, salah satunya adalah serangan DHCP *starvation* [4]. Serangan DHCP

starvation dapat menghabiskan ketersediaan alamat IP pada layanan DHCP di jaringan nirkabel, sehingga perangkat di sisi pengguna tidak dapat melakukan konfigurasi alamat IP secara otomatis dan kemudian gagal terkoneksi ke jaringan internet [5].

Menurut [6] terdapat dua kategori teknik serangan DHCP *starvation*, yaitu: 1) Tipe-1, penyerang mengirimkan paket DHCP *Discover* yang telah dimanipulasi dan dikirimkan sebanyak-banyaknya kepada DHCP *server*, sehingga seakan-akan ada banyak perangkat pengguna yang hendak meminta alamat IP dan hal ini mengakibatkan ketersediaan alamat IP pada DHCP *server* habis dan tidak dapat melayani perangkat yang lain. 2) Tipe-2, penyerang mengirimkan paket ARP (*Address Resolution Protocol*) *reply* palsu setiap kali DHCP *server* mengirimkan paket ARP *request* dalam rangka melakukan pengecekan terhadap eksistensi alamat IP yang akan ditawarkannya, setiap kali DHCP *server* menerima respon berupa paket ARP *reply*, maka alamat IP tersebut dianggap sudah digunakan dan tidak jadi ditawarkan. Jika hal ini terus terjadi akan menghabiskan ketersediaan alamat IP pada DHCP *server* dan tidak bisa melayani perangkat yang lain.

Secara garis besar, metode deteksi dan pencegahan serangan DHCP *starvation* yang telah diajukan oleh para peneliti sebelumnya antara lain: 1) Metode deteksi menggunakan fitur pada perangkat switch *layer* 2 yang diajukan oleh [7] dan [8]. 2) Metode deteksi menggunakan kriptografi yang diajukan oleh [9] dan [10]. 3) Metode deteksi berbasis protokol ICMP (*Internet Control Message Protocol*) yang diajukan oleh [11]. 4) Metode deteksi berbasis *Machine Learning* yang diajukan oleh [12]. Dari semua metode yang telah diajukan oleh peneliti sebelumnya masih memberikan peluang penelitian lebih lanjut dalam hal hasil deteksi yang lebih efektif dan juga penggunaan sumber daya perangkat yang lebih efisien.

Penelitian ini fokus pada merumuskan solusi untuk mengatasi masalah serangan DHCP *starvation* Tipe-2. Untuk meningkatkan efektivitas deteksi serangan DHCP *starvation*, maka digunakan metode deteksi berbasis analisis protokol saat terjadi komunikasi antara DHCP *client* dan DHCP *server*. Hasil analisis tersebut akan menjadi dasar dalam menyusun algoritma deteksi serangan. Hal ini dilakukan agar proses deteksi menjadi lebih spesifik sehingga hasil deteksi jauh lebih akurat. Algoritma deteksi akan diterapkan menggunakan pemrograman Python dan kemudian ditanamkan (*embedded*) ke dalam piranti portabel, seperti penelitian yang dilakukan oleh [13] dan [14] agar penggunaan spesifikasi perangkat menjadi lebih efisien. Piranti portabel sistem deteksi ini sangat cocok diimplementasikan pada jaringan nirkabel di area sekolah, kampus, perkantoran pemerintah, bahkan di jaringan IoT (*Internet of Things*) berbasis nirkabel yang menggunakan layanan DHCP.

2. TINJAUAN PUSTAKA

Penelitian terkait deteksi dan pencegahan serangan DHCP *starvation* telah dilakukan oleh beberapa peneliti sebelumnya, tentu saja dengan pendekatan yang berbeda dan hasil yang berbeda pula.

Penelitian terkait keamanan protokol ARP dan DHCP dilakukan oleh [9] dengan fokus melakukan pencegahan terhadap serangan ARP *spoofing* dan DHCP *starvation*. Metode yang digunakan adalah dengan menerapkan proses autentikasi antara DHCP *client* dan DHCP *server*. Proses autentikasi ini melibatkan server KDA (*Key Distribution and Authentication*). *Server* KDA yang digunakan dalam penelitian memiliki spesifikasi prosesor Core i7 3,6 GHz dan RAM 8 GB, sistem operasi yang digunakan di sisi *client* dan *server* adalah sistem operasi Linux Ubuntu 14.04. *Server* KDA ini akan melakukan validasi terhadap kode ID yang melekat pada paket DHCP *Discover* yang dikirimkan oleh DHCP *client*, jika kode ID tersebut terdaftar pada *server* KDA, maka paket DHCP *Discover* dari DHCP *client* diterima dan direspon oleh DHCP *server*, teknik yang sama juga diterapkan untuk pencegahan terhadap serangan ARP *spoofing*. Kelebihan dari metode ini adalah mampu melakukan pencegahan terhadap serangan DHCP *starvation* dan ARP *spoofing*, namun memiliki kelemahan dari sisi implementasinya di kondisi nyata. Dibutuhkan perubahan protokol di sisi perangkat pengguna dan harus menyiapkan *server* KDA untuk kebutuhan autentikasi, hal ini tidak mudah dilakukan karena beragamnya sistem dan jenis perangkat yang digunakan oleh pengguna.

Penelitian [9] dilanjutkan oleh [12] untut mendeteksi serangan DHCP *starvation* Tipe-2 yang ditemukan oleh [6] sebelumnya saat menemukan celah keamanan fitur deteksi konflik alamat IP pada komunikasi protokol DHCP. Metode yang digunakan adalah deteksi anomali (*Anomaly Based*) berbasis *Machine Learning*. Dataset dibentuk dari hasil tangkapan paket (*packet capture*) DHCP *Discover* baik berupa paket normal maupun paket serangan, paket-paket ini dikumpulkan untuk durasi waktu tertentu. Paket-paket DHCP *Discover* tersebut kemudian diolah dan diberi label, kemudian diuji menggunakan beberapa jenis algoritma klasifikasi berbasis *Machine Learning*. Hasil dari beberapa jenis algoritma klasifikasi menunjukkan bahwa algoritma K-Means *Clustering* memberikan hasil deteksi dengan tingkat akurasi paling tinggi, yaitu sebesar 96,44%. Perangkat yang digunakan dalam proses mengolah dataset dan pengujian adalah komputer dengan prosesor Core i5 quad core dan RAM 16 GB. Kelemahan dari metode deteksi berbasis anomali adalah prosesnya yang rumit dan butuh spesifikasi komputer yang tinggi, sehingga hal ini sulit diterapkan pada piranti portabel.

Penelitian serupa dilakukan oleh [15] terkait deteksi dan pencegahan terhadap serangan DHCP *starvation*. Metode yang digunakan adalah memberikan label pada setiap paket DHCP *Discover* menggunakan kode DSCP (*Differentiated Services Code Point*) yang sebelumnya telah terdaftar alamat MAC dari perangkat yang digunakan untuk terkoneksi ke jaringan. Jika alamat MAC dari perangkat yang terkoneksi tidak terdaftar maka akan diberi kode DSCP dengan angka 0, sedang jika sudah terdaftar akan diberi kode DSCP angka 50. Paket-paket DHCP *Discover* dengan kode DSCP angka 0 akan ditandai sebagai serangan dan langsung dibuang (*drop*) oleh perangkat filter, sedang yang berkode DSCP angka 50 akan diproses lebih lanjut dengan membatasi jumlah maksimal 5 paket saja dalam waktu 1 jam, jika melebih 5 paket dalam 1 jam maka akan ditandai sebagai serangan. Perangkat yang digunakan adalah sebuah piranti portabel dengan spesifikasi prosesor MIPS 400 MHz dan RAM 64 MB. Metode yang digunakan efektif pada area jaringan nirkabel dengan jumlah pengguna yang kecil dan menyediakan seorang admin yang bertugas untuk mendaftarkan alamat MAC dari setiap perangkat yang akan terhubung secara legal. Kelemahan metode ini tidak efektif digunakan pada area jaringan nirkabel yang bersifat publik dengan jumlah pengguna yang banyak. Kelemahan lain dari metode ini adalah belum mampu melakukan deteksi dan pencegahan terhadap serangan DHCP *starvation* Tipe-2.

Peneliti [16] menggunakan metode deteksi dengan pendekatan berbeda, yaitu dengan memanfaatkan protokol ICMP dan ARP untuk melakukan validasi apakah paket DHCP yang dikirimkan oleh client menuju DHCP server berasal dari client yang sah (*legitimate*) ataukah dari penyerang (*malicious*). pada penelitin tersebut digunakan aplikasi simulator Cisco Packet Tracer untuk melakukan simulasi. Metode deteksi yang diterapkan menambahkan protokol ARP dari penelitian sebelumnya [11] yang hanya menggunakan protokol ICMP. Protokol ARP ditambahkan karena protokol ICMP memiliki kelemahan saat *firewall* diterapkan di sisi *client*, yang mempengaruhi validitas hasil deteksi. Kelemahan lain dari metode yang digunakan dalam penelitian ini hanya sebatas melakukan simulasi dan tidak melakukan pengujian pada kondisi real di lapangan, serta belum menerapkan pencegahan terhadap serangan DHCP *starvation*.

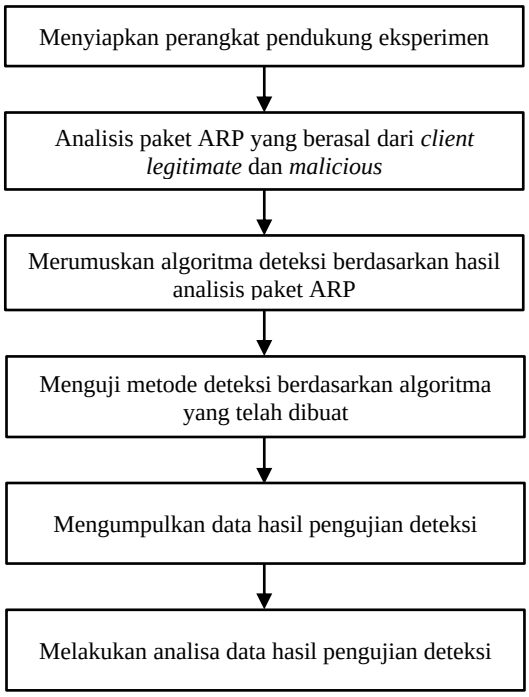
Peneliti [17] melajutkan penelitian yang telah dilakukan oleh [16], metode yang digunakan untuk melakukan deteksi serangan DHCP *starvation* adalah dengan melakukan *port scanning* terhadap *client* yang telah melakukan permintaan alamat IP ke DHCP *server*, Jika alamat port untuk layanan tertentu terdeteksi maka dianggap sebagai *client* sah (*legitimate client*) namun jika tidak terdeteksi alamat *port* tertentu maka dianggal *malicious client*. Dalam proses pengujiannya menggunakan aplikasi simulator EVE-NG. Metode ini memiliki kelemahan akan menghasilkan hasil deteksi yang keliru karena dua hal, yaitu 1) *Firewall* yang aktif di sisi *client* sehingga *port* tertentu tersebut tidak dapat dideteksi, sehingga *client legitimate* dapat terdeteksi sebagai *client malicious* dan demikian sebaliknya. 2) Tidak semua perangkat di sisi *client* mengaktifkan port pada layanan tertentu, contohnya *client* berupa *Smartphone* yang tidak dibekali dengan layanan seperti pada perangkat komputer pada umumnya, sehingga

bisa dipastikan *client* tersebut akan dideteksi sebagai *client malicious*.

Metode yang diusulkan dalam penelitian ini adalah deteksi serangan DHCP *starvation* Tipe-2 menggunakan sistem deteksi dan pencegahan berbasis analisis protokol (*protocol-based analysis*), yang diterapkan pada piranti portabel menggunakan pemrograman Python. Metode deteksi dengan analisis berbasis protokol memiliki kelebihan dalam hal mampu mengenali secara jenis serangan DHCP *starvation* Tipe-2 secara spesifik, sehingga diharapkan tingkat akurasi deteksinya lebih tinggi dari metode-metode yang telah digunakan sebelumnya. Piranti portabel digunakan sebagai mesin IDPS agar penggunaan spesifikasi perangkat fisik menjadi lebih efisien, sehingga dapat diterapkan pada area jaringan nirkabel skala kecil sekalipun, karena tidak membutuhkan biaya yang tinggi

3. METODE PENELITIAN

Tahapan penelitian yang dilakukan mengikuti langkah-langkah seperti pada gambar 1. Langkah pertama adalah menyiapkan semua perangkat pendukung eksperimen yang dibutuhkan.



Gambar 1. Tahapan penelitian

Pada tabel 1 terlampir daftar perangkat eksperimen, diantaranya adalah Laptop *client* yang berjumlah 11 unit dan dibagi menjadi dua kategori, yaitu 1 unit laptop berperan sebagai *client malicious* (penyerang) dengan sistem operasi Linux, 10 unit laptop yang lain digunakan sebagai *client legitimate*, menggunakan sistem operasi Windows 10 dengan alamat IP statis. Smartphone berjumlah 10 unit digunakan sebagai *client* korban, menggunakan sistem operasi Android dengan alamat IP dinamis yang nantinya akan meminta alamat IP dari DHCP *server*. Piranti portabel yang digunakan sebagai sistem IDPS adalah Orange Pi dengan sistem operasi OpenWRT (sistem operasi Linux untuk perangkat portabel). DHCP *server* dan *Bridge Filter* menggunakan perangkat RouterBoard MikroTik.

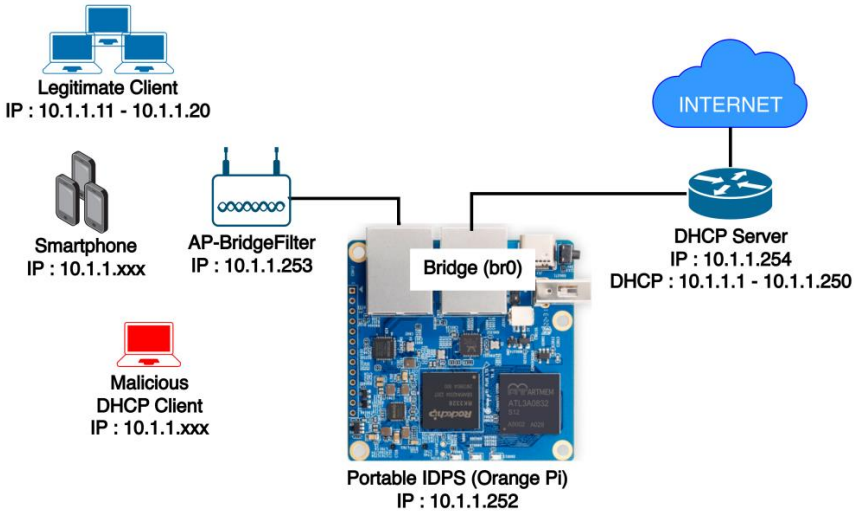
Tabel 1. Perangkat pendukung eksperimen

Jenis Perangkat	Spesifikasi	Jumlah
1. Laptop	Core i3 2GHz, RAM	11 Unit
2. Smartphone	4GB	10 Unit
3. IDPS Portabel	Android	1 Unit
	Orange Pi Prosesor	1 Unit
4. DHCP Server	Quad-Core ARM RAM	1 Unit
5. Bridge Filter	1 GB	2 unit
6. Kabel Jaringan	MikroTik RB951Ui-2n	

Semua instrumen eksperimen tersebut kemudian saling dikoneksikan sesuai dengan topologi jaringan yang terlampir pada gambar 2.

Langkah kedua adalah menangkap (*capture*) paket ARP yang berasal dari *client legitimate* dan *malicious* menggunakan aplikasi Packet Analyzer Wireshark untuk dianalisis agar dapat diketahui perbedaan karakteristik dari kedua jenis paket ARP tersebut. Pada mesin client *malicious* dijalankan aplikasi untuk menjalankan serangan DHCP *starvation* Tipe-2 agar nantinya saat *client* korban meminta alamat IP ke DHCP *server*, maka mesin *client malicious* akan merespon dengan paket ARP *reply* palsu. Alamat IP yang akan diminta ke DHCP *server* oleh *client* korban adalah alamat IP yang digunakan oleh *client legitimate*, sehingga *client legitimate* juga akan merespon dengan paket ARP *reply*.

Langkah ketiga adalah merumuskan algoritma deteksi serangan DHCP *starvation* Tipe-2 berdasarkan hasil analisis paket ARP



Gambar 2. Topologi jaringan eksperimen

yang berasal dari *client legitimate* dan *malicious* di tahap sebelumnya. Algoritma deteksi tersebut nantinya akan diterjemahkan ke dalam bahasa pemrograman Python dengan dukungan pustaka Scapy dan Paramiko. Program deteksi yang telah dibuat kemudian ditanamkan ke piranti portabel Orange Pi.

Langkah keempat adalah melakukan pengujian metode deteksi serangan DHCP *starvation* Tipe-2 berdasarkan algoritma deteksi yang telah ditanamkan ke piranti portabel Orange Pi. Ada dua hal yang akan diuji pada tahap ini, yaitu 1) Menguji apakah sistem deteksi berhasil membedakan antara paket ARP *reply* yang berasal dari *client legitimate* dan *malicious* dan 2) Menguji apakah sistem deteksi juga mampu melakukan pencegahan terhadap serangan yang dilakukan oleh client malicious sehingga client korban kemudian bisa mendapatkan alamat IP dari DHCP *server*. Prosen pengujian dilakukan pada jaringan nirkabel SMKN 2 Palopo.

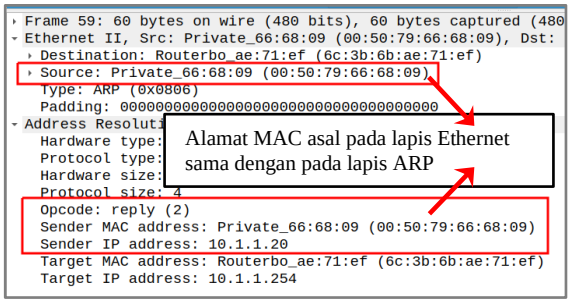
Langkah kelima adalah mengumpulkan data hasil pengujian deteksi. Data yang dikumpulkan dibagi menjadi 4 (empat) kategori: 1) Data *True Positive* (TP), yaitu data jumlah paket ARP *reply* yang berasal dari *client malicious* dan dideteksi sebagai serangan, 2) Data *True Negative* (TN), yaitu data jumlah paket ARP *reply* yang berasal dari *client legitimate* dan tidak terdeteksi sebagai sebagai serangan, 3) Data *False Positive* (FP), yaitu data jumlah paket ARP *reply* yang berasal dari *client legitimate* namun terdeteksi sebagai serangan, dan 4) Data *False Negative* (FN), yaitu data jumlah paket ARP *reply* yang berasal dari *client malicious* namun tidak terdeteksi sebagai serangan.

Langkah keenam adalah melakukan analisis terhadap data yang telah dikumpulkan sebelumnya. Secara umum untuk menghitung tingkat akurasi deteksi menggunakan *Confusion Matrix* [18]. Dari empat kategori data yang telah dikumpulkan sebelumnya, selanjutnya akan digunakan untuk menghitung tingkat akurasi deteksi menggunakan rumus berikut.

$$Akurasi = \frac{(TP + TN)}{(TP + TN + FP + FN)} \times 100\% \quad (1)$$

4. HASIL DAN PEMBAHASAN

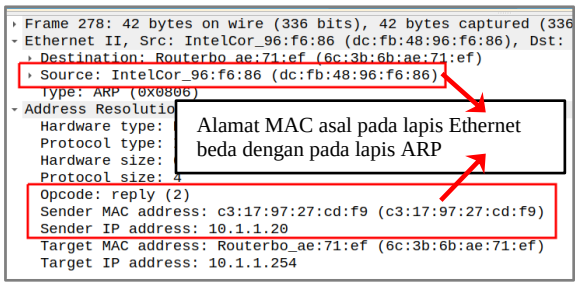
Gambar 3 menunjukkan hasil tangkapan paket ARP *reply* yang berasal dari client legitimate. Fokus perhatian terletak pada lapisan Ethernet dan ARP yang diberi tanda kotak warna merah. Nampak bahwa alamat MAC asal yang tertera pada kedua lapisan tersebut memiliki alamat yang sama.



Gambar 3. Paket ARP *reply* dari *client legitimate*

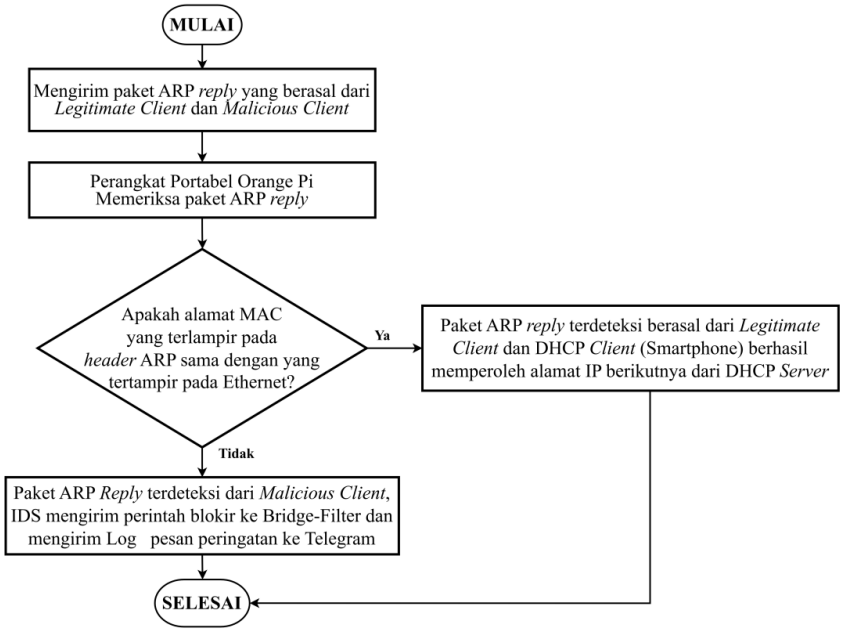
Gambar 4 menunjukkan hasil tangkapan paket ARP *reply* yang berasal dari *client malicious*. Nampak bahwa alamat MAC asal yang tertera

pada kedua lapisan tersebut memiliki alamat yang berbeda. Hal ini menjadi pembeda dari paket ARP *reply* yang dikirimkan oleh *client legitimate*.



Gambar 4. Paket ARP *reply* dari *client malicious*

Berdasarkan dari hasil analisis tangkapan paket ARP *reply* yang berasal dari *client legitimate* dan *malicious* tersebut kemudian disimpulkan bahwa perbedaan karakteristik dari keduanya terletak pada persamaan atau perbedaan antara alamat MAC pada lapisan Ethernet dan ARP. Jika alamat MAC yang terlampir pada lapisan Ethernet sama dengan yang terlampir pada lapisan ARP maka akan dianggap sebagai paket yang sah yang berasal dari *client legitimate*, namun jika berbeda maka akan dianggap sebagai paket serangan yang berasal dari *client malicious*.



Gambar 5. Algoritma deteksi serangan DHCP *starvation* Tipe-2

Dari hasil analisis paket ARP sebelumnya kemudian dirumuskan metode deteksi yang akan diterapkan. Gambar 5 menunjukkan metode deteksi yang akan diterapkan berdasarkan hasil analisis paket ARP sebelumnya. Algoritma deteksi ini kemudian diterjemahkan ke pemrograman Python yang ditanamkan pada piranti portabel Orange Pi. Fungsi pencegahan serangan ditambahkan dengan melakukan pemblokiran akses dari penyerang dan mengirimkan pesan peringatan ke pengelola lewat layanan bot telegram.

Gambar 6 menunjukkan potongan kode program Python yang digunakan untuk mendeteksi paket ARP *reply* yang berasal dari client legitimate, sedang pada Gambar 7 adalah potongan kode program untuk mendeteksi paket ARP *reply* yang berasal dari *client malicious*. Kode program Python lengkap dapat diunduh pada alamat tautan bit.ly/dhcpidpsbahar.

Selanjutnya aplikasi penyerang di sisi *client malicious* dijalankan, dan kemudian aplikasi sistem IDPS juga dijalankan pada piranti portabel Orange Pi, para *client legitimate* pada posisi *standby* menunggu respon dari DHCP server saat menawarkan kepada Smartphone alamat IP yang sama digunakan oleh para *client legitimate* tersebut. Smartphone sebagai *client* korban melakukan koneksi ke jaringan nirkabel dan mulai meminta alamat MAC kepada DHCP *server*. Kemudian DHCP *server*

```
#Jika alamat MAC pada lapisan Ethernet dan ARP sama
#Maka akan terdeteksi sebagai Legitimate Client
if pkt[Ether].src == pkt[ARP].hwsrc:
   print("="*54)
   print("\nTerdeteksi Legitimate Client " + pkt[Ether].src)
   print("Menggunakan alamat IP Statis " + pkt[ARP].psrc)
   pesan_valid()
   print("="*54)
```

Gambar 6. Kode program untuk deteksi *client legitimate*

```
#Jika alamat MAC berbeda maka akan terdeteksi sebagai serangan
else:
   print("="*54)
   print("\nTerdeteksi Malicious Client " + pkt[Ether].src)
   print("Membuat konflik pada IP " + pkt[ARP].psrc)
   blokir_akses()
   pesan_attack()
   print("="*54)
```

Gambar 7. Kode program untuk deteksi *client malicious*

akan merespon dengen mengirimkan paket ARP request ke jaringan. Aplikasi di sisi *client malicious* akan segera merespon dengan mengirimkan paket ARP *reply* palsu dengan tujuan untuk membuat konflik terhadap permintaan alamat IP 10.1.1.20 di DHCP *server* (Gambar 8). Karena alamat IP yang ditawarkan oleh DHCP *server* sama dengan alamat IP yang digunakan oleh *client legitimate*, maka *client legitimate* juga akan merespon dengan mengirimkan paket ARP reply. Sistem IDPS akan memeriksa paket ARP *reply* yang berasal dari *client legitimate* dan *malicious* dan kemudian menentukan status paket ARP *reply* tersebut apakah masuk kategori paket sah (*legitimate*) ataukan paket *malicious*.

```
:~/python$ sudo ./dhcpconflict.py

Waiting for ARP Request from DHCP Server...


Receiving ARP Request for IP 10.1.1.20
Sending IP Conflict packet to DHCP Server...
```

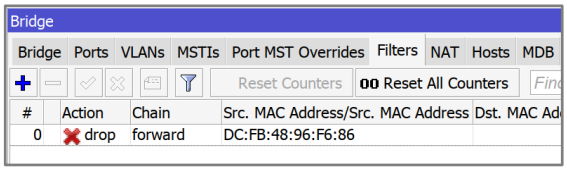
Gambar 8. *Client malicious* melakukan penyerangan ke DHCP *server*

Gambar 9 adalah hasil deteksi sistem IDPS terhadap paket ARP *reply* sebelumnya. Nampak bahwa paket-paket ARP *reply* tersebut berhasil diidentifikasi, baik yang berasal dari *client legitimate* dengan alamat MAC asal "00:50:79:66:68:09" maupun *client malicious* dengan alamat MAC asal "DC:FB:48:96:F6:86" . Dalam proses pencegahan, maka sistem IDPS akan menggunakan alamat MAC sumber yang tertera pada lapisan Ethernet sebagai referensi pemblokiran.

```
root@dhcpidps:~# ./dhcpids.py

##############################################
## SELAMAT DATANG DI SISTEM IDPS SMKN2PALOPO ##
##############################################


====================================================

Terdeteksi Legitimate Client 00:50:79:66:68:09
Menggunakan alamat IP Statis 10.1.1.20
Pesan peringatan telah dikirim ke Admin
====================================================
====================================================

Terdeteksi Malicious Client dc:fb:48:96:f6:86
Membuat konflik pada IP 10.1.1.20
Mengirim perintah Blokir ke Bridge-Filter

Pesan peringatan telah dikirim ke Admin
====================================================
```

Gambar 9. Hasil deteksi sistem IDPS

Perintah blokir oleh sistem IDPS dikirimkan ke perangkat *Bridge Filter* (Gambar 10), nampak bahwa pada *Bridge Filter* telah ditambahkan satu baris aturan blokir (*drop*) yang akan membuat *client malicious* tidak dapat lagi terkoneksi ke jaringan.



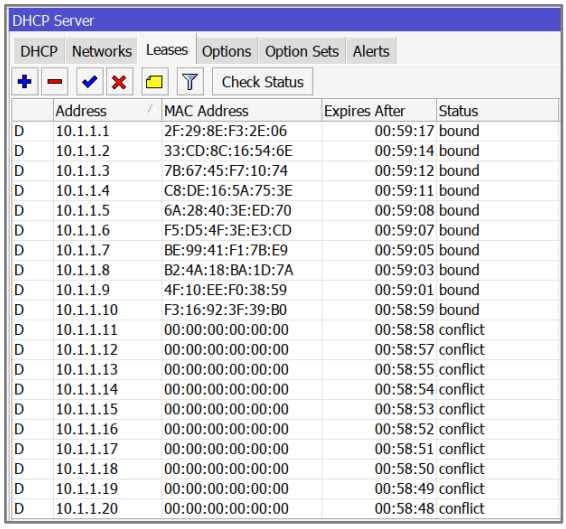
Gambar 10. Blokir akses *client malicious*

Setelah itu dikirimkan pula pesan peringatan lewat layanan bot Telegram (Gambar 11). Pesan peringatan ini sebagai informasi kepada administrator jaringan agar dapat diketahui waktu dan asal serangan terhadap DHCP *server*.

Gambar 11. Pesan dikirimkan ke Telegram

Setelah sistem IDPS bekerja, tampak bahwa konflik alamat IP hanya terjadi pada alamat-

alamat IP yang telah digunakan oleh *client legitimate* saja dan ini merupakan hal yang wajar terjadi. Saat terjadi konflik alamat IP dan DHCP *server* batal menawarkan alamat IP yang telah digunakan oleh *client legitimate* maka Smartphone akan mengulangi permintaan alamat IP kepada DHCP *server*, bisa diperhatikan pada gambar 12 nampak bahwa pada alamat IP 10.1.1.1 hingga 10.1.1.10 berhasil ditawarkan dan digunakan oleh para Smartphone dengan status "*bound*". Pada kondisi sebelum diterapkan IDPS, maka semua persediaan alamat IP akan berstatus "*conflict*", sehingga membuat semua Smartphone gagal memperoleh alamat IP dari DHCP *server* dan menyebabkan tidak dapat terkoneksi ke jaringan. Dari total 249 alamat IP yang tersedia, 10 alamat IP diantaranya tidak akan ditawarkan karena telah digunakan oleh *client legitimate*, sehingga tersisa sebanyak 239 alamat IP yang bisa digunakan oleh para *client* Smartphone.



	Address	MAC Address	Expires After	Status
D	10.1.1.1	2F:29:8E:F3:2E:06	00:59:17	bound
D	10.1.1.2	33:CD:8C:16:54:6E	00:59:14	bound
D	10.1.1.3	7B:67:45:F7:10:74	00:59:12	bound
D	10.1.1.4	C8:DE:16:5A:75:3E	00:59:11	bound
D	10.1.1.5	6A:28:40:3E:ED:70	00:59:08	bound
D	10.1.1.6	F5:D5:4F:3E:E3:CD	00:59:07	bound
D	10.1.1.7	BE:99:41:F1:7B:E9	00:59:05	bound
D	10.1.1.8	B2:4A:18:BA:1D:7A	00:59:03	bound
D	10.1.1.9	4F:10:EE:F0:38:59	00:59:01	bound
D	10.1.1.10	F3:16:92:3F:39:B0	00:58:59	bound
D	10.1.1.11	00:00:00:00:00:00	00:58:58	conflict
D	10.1.1.12	00:00:00:00:00:00	00:58:57	conflict
D	10.1.1.13	00:00:00:00:00:00	00:58:55	conflict
D	10.1.1.14	00:00:00:00:00:00	00:58:54	conflict
D	10.1.1.15	00:00:00:00:00:00	00:58:53	conflict
D	10.1.1.16	00:00:00:00:00:00	00:58:52	conflict
D	10.1.1.17	00:00:00:00:00:00	00:58:51	conflict
D	10.1.1.18	00:00:00:00:00:00	00:58:50	conflict
D	10.1.1.19	00:00:00:00:00:00	00:58:49	conflict
D	10.1.1.20	00:00:00:00:00:00	00:58:48	conflict

Gambar 12. Kondisi DHCP *server* setelah IDPS diterapkan

Untuk mengukur tingkat akurasi deteksi dan pencegahan, dibutuhkan empat komponen penting, yaitu mengukur nilai *True Positive* (TP), *True Negative* (TN), *False Positive* (FP) dan *False Negative* (FN). Proses pengukuran akan dilakukan sebanyak empat kali dengan kondisi yang sama, mesin *client legitimate* terdiri dari 10 unit komputer sehingga akan mengirimkan sejumlah 10 paket ARP *reply* kepada DHCP *server* dan semua paket ARP *reply* tersebut berhasil dideteksi oleh IDPS. Karena proses pengukuran dilakukan sebanyak empat kali, maka total paket yang dikirimkan *client legitimate* adalah 40 paket ARP *reply*. Mesin *client malicious* akan mengirimkan sebanyak 249 paket ARP *reply* palsu yang dialokasikan untuk membuat konflik 249 alamat IP pada DHCP *server*, dan total 996 paket ARP *reply* palsu yang berasal dari *client malicious* berhasil dideteksi oleh IDPS. Hasil pengukuran dapat dilihat pada Tabel 2, selain mengukur jumlah keempat komponen tersebut, juga diukur kecepatan deteksi dari sistem IDPS, untuk mengetahui seberapa cepat sistem IDPS merespon adanya serangan terhadap DHCP *server*, tampak pada tabel pengukuran bahwa rerata kecepatan deteksi sistem IDPS adalah sebesar 55 ms. Hasil pengukuran ini nantinya akan digunakan dalam proses menghitung tingkat akurasi sistem deteksi IDPS.

Tabel 2. Hasil pengukuran sistem deteksi IDPS

Iterasi	TP	TN	FP	FN	Kecepatan Deteksi
1	249	10	-	-	52 ms
2	249	10	-	-	55 ms
3	249	10	-	-	50 ms
4	249	10	-	-	63 ms
Total	996	40	-	-	220 ms
Rerata	249	10	-	-	55 ms

Tahap selanjutnya adalah memasukkan hasil pengukuran ke dalam rumus perhitungan tingkat akurasi deteksi. Hasil perhitungan menunjukkan bahwa tingkat akurasi deteksi dapat mencapai 100%.

$$Akurasi = \frac{(TP + TN)}{(TP + TN + FP + FN)} \times 100\%$$

$$Akurasi = \frac{(996 + 40)}{(996 + 40 + 0 + 0)} \times 100\%$$

$$Akurasi = 100\%$$

Untuk mengetahui posisi penelitian yang telah dilakukan terhadap penelitian sebelumnya, maka perlu dibandingkan dengan hasil penelitian terdahulu yang terkait dengan topik yang diteliti. Tampak pada tabel 3 bahwa penelitian yang dilakukan memiliki kemampuan deteksi dan pencegahan yang lebih baik, demikian dengan efisiensi penggunaan sumber daya perangkat yang digunakan untuk melakukan deteksi dan pencegahan serangan DHCP *starvation*.

Tabel 3. Perbandikan sistem deteksi dan pencegahan dengan penelitian terkait

Peneliti	Efektivitas	Efisiensi
Younes (2017)	Mampu mendeteksi dan mencegah serangan DHCP *starvation* Tipe-2	Butuh perangkat fisik deteksi dengan spesifikasi tinggi
Tripathi & Hubballi (2018)	Hanya mampu mendeteksi serangan DHCP *starvation* Tipe-2 dengan akurasi 96,44%	Butuh perangkat fisk deteksi dengan spesifikasi tinggi
Sarip & Setyanto (2019)	Tidak mampu mendeteksi serangan DHCP *starvation* Tipe-2	Menggunakan perangkat fisik deteksi spesifikasi rendah
Yaibuates & Chaisricharoen (2020)	Mampu mendeteksi serangan DHCP *starvation* Tipe-2 namun dapat menimbulkan *False Positive* dan *False Negative* akibat *Firewall* di sisi *client*	Menggunakan *software* aplikasi simulator Cisco Packet Tracer
Jony et. al (2023)	Mampu mendeteksi serangan DHCP *starvation* Tipe-2 namun dapat menimbulkan *False Positive* dan *False Negative* akibat *Firewall* dan ketersediaan layanan di sisi *client*	Menggunakan *software* aplikasi simulator EVE-NG
Bahar et. al (2024)	Mampu mendeteksi dan mencegah serangan DHCP starvation Tipe-2 dengan akurasi 100%	Menggunakan perangkat portabel dengan spesifikasi rendah

5. KESIMPULAN

Dari hasil penelitian yang telah dilakukan dapat disimpulkan bahwa serangan DHCP *starvation* Tipe-2 ini dapat dicegah dengan tingkat akurasi 100% menggunakan metode deteksi berbasis analisis protokol ARP. Tingkat akurasi yang tinggi dalam proses deteksi dimungkinkan karena kelebihan dari metode deteksi berbasis protokol adalah kemampuannya mengenali jenis serangan secara spesifik. Dari sisi efisiensi, perangkat yang digunakan dalam penelitian ini menggunakan perangkat fisik portabel dengan spesifikasi rendah, sehingga solusi yang ditawarkan dapat diterapkan pada jaringan nirkabel skala kecil sekalipun, misalnya jaringan nirkabel pada area kantor atau sekolah.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada pihak-pihak terkait yang telah memberi dukungan terhadap penelitian ini.

DAFTAR PUSTAKA

[1] APJII, “Hasil Survei Profil Internet Indonesia 2022.” [Online]. Available: https://apjii.or.id/content/read/39/559/Hasil-Survei-Profil-Internet-Indonesia-2022.

[2] R. E. Droms and T. Lemon, The DHCP handbook. SAMS Publishing, 2003.

[3] C. Lin, T. Su, and Z. Wang, “Summary of high-availability DHCP service solutions,” in 2011 4th IEEE International Conference on Broadband Network and Multimedia Technology, IEEE, 2011, pp. 12–17.

[4] H. Mukhtar, K. Salah, and Y. Iraqi, “Mitigation of DHCP starvation attack,” Computers & Electrical Engineering, vol. 38, no. 5, pp. 1115–1128, 2012.

[5] N. Tripathi and N. Hubballi, “A probabilistic anomaly detection scheme to detect DHCP starvation attacks,” in 2016 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS), IEEE, 2016, pp. 1–6.

[6] N. Tripathi and N. Hubballi, “Exploiting DHCP server-side IP address conflict detection: A DHCP starvation attack,” in 2015 IEEE International Conference on Advanced Networks and Telecommuncations Systems (ANTS), IEEE, 2015, pp. 1–3.

[7] T. J. OConnor, “Detecting and responding to data link layer attacks,” SANS Institute InfoSec Reading Room, Oct, vol. 13, 2010.

[8] C. Toprak, C. Turker, and A. T. Erman, “Detection of DHCP starvation attacks in software defined networks: a case study,” in 2018 3rd international conference on computer science and engineering (UBMK), IEEE, 2018, pp. 636–641.

[9] O. S. Younes, “Securing ARP and DHCP for mitigating link layer attacks,” Sādhanā, vol. 42, pp. 2041–2053, 2017.

[10] A. Shete, A. Lahade, T. Patil, and R. Pawar, “DHCP protocol using OTP based two-factor authentication,” in 2018 2nd International Conference on Trends in Electronics and

Informatics (ICOEI), IEEE, 2018, pp. 136–141.
[11] M. Yaibuates and R. Chaisricharoen, “ICMP based malicious attack identification method for DHCP,” in The 4th Joint International Conference on Information and Communication Technology, Electronic and Electrical Engineering (JICTEE), IEEE, 2014, pp. 1–5.
[12] N. Tripathi and N. Hubballi, “Detecting stealth DHCP starvation attack using machine learning approach,” Journal of Computer Virology and Hacking Techniques, vol. 14, pp. 233–244, 2018.
[13] C. Nykvist, M. Larsson, A. H. Sodhro, and A. Gurtov, “A lightweight portable intrusion detection communication system for auditing applications,” International Journal of Communication Systems, vol. 33, no. 7, p. e4327, 2020.
[14] V. Visoottiviseth, G. Chutaporn, S. Kungvanruttana, and J. Paisarnduangjan, “PITI: Protecting Internet of Things via Intrusion Detection System on Raspberry Pi,” in 2020 International Conference on Information and Communication Technology Convergence (ICTC), IEEE, 2020, pp. 75–80.
[15] A. Setyanto, “Packet Filtering Based On Differentiated Services Code Point For DHCP Starvation Attacks Prevention,” Pekommas, vol. 4, no. 2, pp. 137–146, 2019.
[16] M. Yaibuates and R. Chaisricharoen, “A combination of ICMP and ARP for DHCP malicious attack identification,” in 2020 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (ECTI DAMT & NCON), IEEE, 2020, pp. 15–19.
[17] A. Jony, A. S. M. Miah, and M. N. Islam, “An Effective Method to Detect DHCP Starvation Attack using Port Scanning,” in 2023 International Conference on Next-Generation Computing, IoT and Machine Learning (NCIM), IEEE, 2023, pp. 1–6.
[18] G. Kumar, “Evaluation metrics for intrusion detection systems-a study,” Evaluation, vol. 2, no. 11, pp. 11–7, 2014.