Pengamanan Objek IoT terhadap Serangan *On-off Attack* Menggunakan Manajemen *Trustworthiness*

Anggi Pratama Nasution[1], Dr. Vera Suryani, S.T., M.T.[2], Aulia Arif Wardana S.Kom., M.T.[3]

[1,2,3]Fakultas Informatika, Universitas Telkom, Bandung
[1]praanggi@students.telkomuniversity.ac.id, [2]verasuryani@telkomuniversity.ac.id,
[3]auliawardan@telkomuniversity.ac.id

Abstrak

Internet of Things (IoT) akan menciptakan dunia di mana benda-benda fisik diintegrasikan dengan *seamlessly network* jaringan informasi dalam rangka memberikan layanan yang canggih dan cerdas untuk kehidupan manusia. Terdapat berbagai bentuk ancaman dan serangan terhadap objek IoT yang dapat menyebabkan penyalahgunaan data atau informasi pada objek IoT tersebut. Salah satu bentuk serangannya adalah *On-off Attack*. Pada jenis serangan ini, penyerang berprilaku sebagai objek yang terkadang bersifat baik dengan mengirimkan nilai *Trust* yang valid namun terkadang bersifat jahat dengan mengirimkan nilai *Trust* yang tidak valid sehingga perlunya pengamanan objek terhadap jenis serangan yang seperti ini. Maka dari itu penulis memanfaatkan manajemen *Trustworthiness* sebagai metode untuk mengatasi masalah tersebut. Manajemen *Trustworhiness* dapat memanfaatkan aspek keamanan nilai kepercayaan atau *Trust Value* sebagai acuan pendeteksian serangan terhadap objek. Selain itu, dengan ditambahnya sistem keamanan mengunakan autentikasi yang disediakan oleh MQTT diharapkan dapat memberikan keamanan tambahan. Adapun pendekatan yang dilakukan pada penelitian ini yaitu pengujian pendeteksian serangan *On-off Attack* langsung pada objek yang terhubung dalam jaringan. Hasil pengujian selanjutnya ditampilkan pada halaman web yang telah dibuat menggunakan php juga penggunaan MySQL database sebagai penyimpanan nilai yang dikirim objek ke server. Pengujian pendeteksian serangan *On-off Attack* berhasil dilakukan dengan tingkat keberhasilan sebesar 100% dan waktu ekekusi hingga pendeteksian selesai didapat selama 0.5518318 detik. Hal ini menunjukan bahwa Manajemen *Trustworthiness* dapat digunakan sebagai salah satu metode untuk menangani serangan *On-off Attack*.

Kata kunci : *Internet of Things*, Objek, *On-off Attack*, *Trustworthiness*, Autentikasi.

Abstract

The Internet of Things (IoT) will create a world where physical objects are seamlessly integrated with a network of information networks in order to provide great and intelligent services for human life. There are various forms of threats and attacks on IoT objects that can cause misuse of data or information on the IoT object. One form of attack is the On-off Attack. In this type of attack, the attacker behaves as an object that is sometimes good by sending a valid Trust value, but sometimes is bad by sending an invalid Trust value so that it is necessary to safeguard the object against this type of attack. Therefore, the authors use Trustworthiness management as a method to overcome these problems. Trustworthiness management can rely on the security aspect of Trust Value as a reference to detect attacks on objects. In addition, the added security system using authentication provided by MQTT is expected to provide additional security. The approach taken in this research is testing the detection of On-off Attack directly on objects connected to the network. The test results are then displayed on a web page that has been created using php also use the MySQL database as a storage of values sent by objects to the server. Testing of detection of attacks On-off Attack was successfully carried out with a success rate of 100% and the execution time until the detection is complete is obtained for 0.5518318 seconds. This shows that Trustworthiness Management can be used as a method to deal with On-off Attack.

Keywords: Internet of Things, Objects, On-off Attack, Trustworthiness, Authentication.

1. Pendahuluan

Latar Belakang

Salah satu tantangan penelitian di bidang IoT adalah masalah keamanan. Jumlah perangkat yang dapat dihubungkan satu sama lain melalui Internet dapat menciptakan potensi besar untuk serangan. Objek pada IoT bisa dimanipulasi untuk proses pengumpulan, penyimpanan, dan analisis data. Data tersebut dibutuhkan oleh

aplikasi kesehatan, pendidikan, transportasi, dan industri untuk pemanfaatan kehidupan manusia. Peningkatan penggunaan data dalam IoT berpotensi menyebabkan beberapa aktivitas serangan dari malicious node, mengingat bahwa aktivitas objek IoT mungkin terbuka untuk siapa saja [1].

Untuk mendeteksi serangan yang dapat terjadi di dalam lingkungan IoT, harus diawali dengan mengenali kemungkinan resiko yang dapat terjadi. Dengan menyediakan objek yang memiliki keamanan yang memadai dapat mengurangi resiko. Mengamankan objek adalah hal penting yang bisa pertama kali dilakukan untuk menjaga proses komunikasi data antar objek [1]. *Trustworthiness Management* adalah salah satu aspek keamanan untuk mengamankan objek IoT. Ini bertujuan untuk meningkatkan kerja sama antara entitas dalam sistem terdistribusi dengan memprediksi perilaku objek di masa depan berdasarkan perilaku sebelumnya [2].

Terdapat berbagai bentuk ancaman dan serangan terhadap objek pada IoT yang dapat menyebabkan penyalahgunaan data pada objek IoT tersebut. Salah satu bentuk serangan pada objek IoT adalah *on-off attack*. Pada jenis serangan ini, objek berprilaku secara acak. Sewaktu-waktu berprilaku sebagai objek yang tidak berbahaya dengan memberikan *real trust value* pada *trustworthiness management*, namun terkadang berprilaku sebagai objek yang berbahaya dengan memberikan *false trust value* pada *trusthiness management* [1].

Berdasarkan hal tersebut dibutuhkan sebuah cara untuk mengamankan objek IoT yaitu menggunakan aspek nilai kepercayaan dari metode *Trustworthiness Management*. Adapun pendekatan yang akan dilakukan yaitu dengan cara implementasi berdasarkan aspek nilai kepercayaan dari *Trustworthiness Management*.

Topik dan Batasannya

Adapun perumusan masalah yang telah dibuat berdasarkan latar belakang diatas yaitu mendeteksi objek yang melakukan serangan *On-off Attack* (*Good-mouthing attacks, Bad-mouthing attacks)* dengan menggunakan manajemen *Trustworthiness* juga menghitung tingkat keberhasilan dan waktu yang dibutuhkan server untuk pendeteksian objek.

Adapun batasan masalah yang dibuat pada penelitian ini yaitu,

1. Keterbatasan kemampuan objek untuk melakukan komputasi algoritma yang kompleks. Penulis disini menggunakan ESP8266 sebagai mikrokontroler yang berperan sebagai *objek* yang terhubung pada server. Penggunakan ESP8266 untuk mengetahui kemampuan alat melakukan implementasi pada penelitian ini.
2. Keterbatasan kemampuan objek untuk melakukan blok pada suatu jaringan sebagai langkah pencegahan terhadap serangan *on-off attack* hingga pada akhirnya tindakan yang dilakukan berupa peringatan yang ditampilkan pada antarmuka web sederhana.
3. Pembuatan jaringan sederhana menggunakan Raspberry pi 3 b+ sebagai server.
4. Penggunaan autentikasi yang sudah disediakan oleh MQTT Broker.

Tujuan

Adapun tujuan yang dibuat berdasarkan rumusan masalah diatas, yaitu untuk mengamankan objek IoT dari serangan *On-off Attacks* (*Bad-mouting attacks, Good-mouting attacks*) dengan menggunakan *Trustworthiness Management* dan juga menghitung tingkat keberhasilan dan waktu yang dibutuhkan server untuk pendeteksian objek.

Organisasi Tulisan

Organisasi penulisan dalam penulisan Tugas Akhir ini yaitu sudah dijelaskan pada bagian pertama mengenai latar belakang dari penelitian, perumusan masalah dari penelitian, tujuan dari penelitian, batasan masalah dari penelitian, dan organisasi penulisan. Pada bagian dua dijelaskan penelitian sebelumnya dan landasan teori yang dapat mendukung penelitian. Pada bagian tiga dijelaskan gambaran umum sistem dan skenario pengujian yang telah dibuat. Pada bagian empat dijelaskan hasil pengujian yang telah dilakukan. Pada bagian lima dijelaskan kesimpulan dan saran dari hasil penelitian yang telah dilakukan.

2. Studi Terkait

Beberapa jurnal dan paper terkait yang berhubungan dengan penelitian ini.

Pada paper ini [1] melakukan sebuah penelitian tentang pendektesi serangan *On-off Attacks* terhadap objek IoT. Adapun pendekatan yang dilakukan yaitu dengan simulasi. Proses deteksi dilakukan dengan menandai objek yang terdeteksi yang melakukan serangan ini: *good-mouthing attacks*, *bad-mouthing attacks*, dan *ballot-stuffing attacks*. Objek kemudian diberi kesempatan untuk berperilaku sebagai objek yang tidak berbahaya, dan dilarang melakukan serangan lagi. Tetapi jika objek yang ditandai masih melakukan beberapa serangan di selanjutnya, maka objek tersebut akan dihukum dengan ditunda waktu layanannya dan diblokir dari komunitas. Metode penandaan ini adalah sarana untuk mendeteksi serangan on-off, dan sangat layak untuk diterapkan pada objek.

Pada paper ini [2] melakukan sebuah penelitian yaitu menerapkan konsep Predictability Trust. Ini adalah sebuah konsep yang memungkinkan akumulasi perilaku sebelumnya untuk menghitung kepercayaan sebuah node dalam suatu sistem. Dengan menggunakan Predictability Trust, perilaku penyerang sebelumnya akan lebih waspada pada saat melakukan penyerangan. Jika perilaku yang terakumulasi terbukti cukup buruk, kolaborator akan memilih untuk tidak bekerja dengan penyerang lagi. Pada penelitian ini, penulis melakukan pendekatan dengan cara menerapkan pada Sliding Window.

Pada paper ini [3] melakukan sebuah penelitian yaitu simulasi dari Trust-based attacks terhadap Internet of Things berhasil dipecahkan dengan menggunakan ConTrust model. Dari hasil simulasi yang didapat disimpulkan bahwa model ConTrust terbukti menjadi model yang stabil dibandingkan dengan model ini.

Pada paper ini [4] dijelaskan *trust management* baru dan *redemption scheme* yang dapat membedakan antara *temporary errors* dan *bad behaviours*. Di sini disajikan skema *trust management* yang efisien dan fleksibel yang mendeteksi dan menahan terhadap serangan *On-off Attacks*. *Predictability Trust* berfungsi untuk mendeteksi *On-off Attacks*. Penulis menggunakan pendekatan dengan Sliding Window untuk melacak perilaku sebelumnya sehingga dapat menentukan seberapa cepat untuk mengembalikan *trust value*. Dengan bantuan Sliding Window perilaku node dalam suatu sistem dapat dianalisis sehingga sesuai dengan sifat perilaku node dapat dikategorikan sebagai normal atau berbahaya. Sliding Window dapat menghitung nilai *good behaviours* dan *bad behaviours* dari setiap node dalam sistem. Sistem membantu mendeteksi dan mencegah *On-off Attacks* secara efektif dengan menghitung kepercayaan setiap node dalam sistem.

Pada paper ini [5], dilakukan sebuah penelitian untuk mendeteksi *On-off* attacks metode *smart trust management* berdasarkan *machine learning* dan Sliding Window. Metode keamanan yang diusulkan mampu mendeteksi *On-off Attacks* di IoT dengan 97% presisi dalam dataset dunia nyata dan 96% presisi dalam lingkungan simulasi. Dibandingkan dengan penelitian lain, metode ini adalah 95% lebih cepat dalam mengidentifikasi *On-off Attacks*.

On-off Attack

On-off Attacks yaitu salah satu bentuk jenis serangan pada objek IoT. Pada jenis serangan ini, objek berprilaku secara acak. Sewaktu-waktu berprilaku sebagai objek yang tidak berbahaya yaitu dengan cara memberikan *real trust value* pada *trust management*, namun terkadang berprilaku sebagai objek yang berbahaya yaitu dengan cara memberikan *false trust value* pada *trust management* [1].

Sebagian besar skema penukaran kepercayaan gagal untuk secara efektif membedakan antara *on-off attack* dan *temporary errors*, terutama ketika penyerang berperilaku baik. Oleh karena itu, penyerang mungkin dapat tetap aktif dalam sistem dengan menyamarkan serangan sebagai *temporary errors* [2][6].

Dalam *On-off Attacks* terdapat 2 kondisi. Pertama yaitu *On state* disebut juga sebagai *attack state*. Ketika sebuah node berbahaya berada dalam keadaan *On state*, node tersebut menyerang node target dengan serangan yang terkait. Lalu selanjutnya yaitu keadaan *Off-state* disebut sebagai keadaan normal. Ketika node berbahaya dalam keadaan *Off*, itu berperilaku normal. Ketika rasio *Off state* ke *On state* tinggi, sistem manajemen kepercayaan mungkin mengalami kesulitan mendeteksi perilaku jahat. Semakin tinggi rasio *Off-to-On*, semakin lama penyerang dapat tetap berada dalam sistem [7].

Trustworthiness Management

Pada penelitian [1] pendeteksian serangan *On-off Attack* dilakukan dua fase. Fase yang pertama yaitu setiap objek harus terautentikasi terlebih dahulu dengan MQTT Broker. Fase kedua yaitu, setiap objek yang terautentikasi selanjutnya dapat melakukan pengiriman nilai ke server melalui MQTT Broker. Nilai tersebut selanjutnya dilakukan pengecekan untuk mengetahui objek tersebut melakukan serangan.

Trust pada IoT adalah istilah yang melibatkan analisis perilaku perangkat yang terhubung ke jaringan yang sama. Hubungan kepercayaan antara dua perangkat membantu dalam mempengaruhi perilaku interaksi mereka di masa depan. Ketika perangkat saling mempercayai, mereka lebih suka berbagi layanan dan sumber daya sampai batas tertentu. *Trust Management* memungkinkan perhitungan dan analisis kepercayaan di antara perangkat untuk membuat keputusan yang sesuai untuk membangun komunikasi yang efisien dan andal di antara perangkat [8]. Pada dasarnya, manajemen kepercayaan adalah mekanisme untuk mengevaluasi, membangun, memelihara, dan mencabut kepercayaan antara perangkat dari jaringan yang sama atau berbeda dalam lingkungan IoT [5].

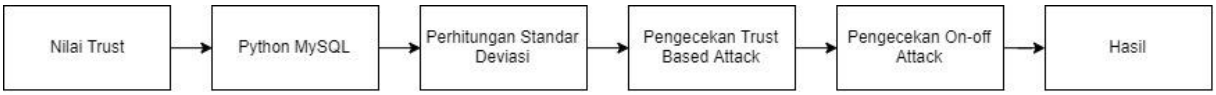
Trustworthiness Management adalah salah satu aspek keamanan untuk mengamankan objek IoT. Ini bertujuan untuk meningkatkan kerja sama antara entitas dalam sistem terdistribusi dengan memprediksi perilaku objek di masa depan berdasarkan perilaku sebelumnya [2]. Dengan menggunakan *Trustworthiness Management* kita dapat mendeteksi serangan *On-off Attack* berdasarkan aspek dari pengiriman nilai kepercayaan / *Trust Value*. Dengan menggunakan nilai *trust* yang telah dihitung sebagai batas dari objek melakukan serangan *On-off Attack*.

Menurut penelitian [9]. Ada beberapa jenis serangan lain yang sering terjadi pada *trust management* di dalam IoT berdasarkan asas kepercayaan / *trust*. Diantaranya sebagai berikut.

1. *Good-Mouthing attacks*: memberikan nilai kepercayaan salah dari suatu objek, yaitu nilai berlebihan
2. *Bad-Mouthing attacks*: memberikan nilai kepercayaan salah dari suatu objek, yaitu menjelekkan suatu objek

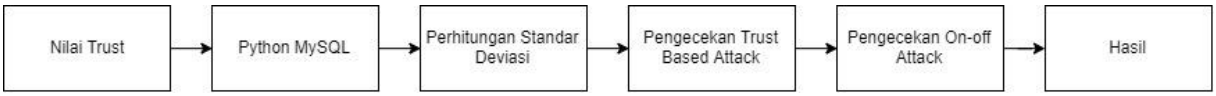
3. Sistem yang Dibangun

Gambaran Umum Sistem



Gambar 1 Gambaran Umum Sistem

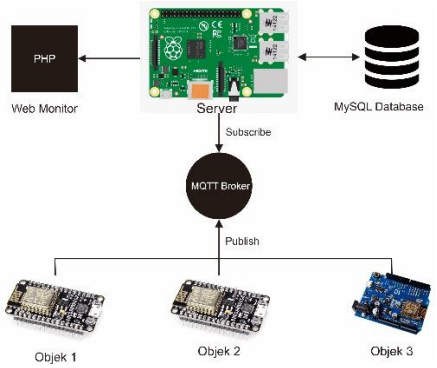
Pada



Gambar 1 diatas dijelaskan alur sistem yang dibangun. Langkah pertama yaitu terdapat tiga buah objek ESP8266 sebagai *objek* yang dapat terhubung pada server Raspberrypi. Masing-masing objek mengirimkan sebuah nilai *trust* terhadap objek lain ke server menggunakan mqtt yang terhubung pada raspberrypi. Tahap selanjutnya nilai *trust* yang diterima oleh raspberrypi disimpan ke dalam MySQL Database. Data yang telah disimpan selanjutnya di olah dengan melakukan perhitangan statistik matematika yaitu standar deviasi. Standar deviasi untuk mendapatkan nilai yang digunakan sebagai batas untuk pengecekan *Trustbased attack*. Pada tahap pengecekan *Trustbased attack* (*Good-mouthing* dan *Bad-mouthing*) dilakukan, masing-masing objek dilakukan pengecekan nilai *trust*. Banyaknya serangan *Good-mouhting* dan *Bad-mouthing* dijumlahkan sebagai *Trustbased attack*. Selanjutnya tahap pengecekan *On-off Attack*, pada tahap ini objek yang terindikasi melakukan *Trustbased attack* dilakukan pengecekan, jika banyaknya serangan *Trustbased attack* pada tiap objek lebih dari dari tiga serangan maka objek tersebut dikatakan melakukan serangan *On-off Attack*. Hasil dari pendeteksian *On-off Attack* selanjutnya ditampilkan pada antarmuka web. Pada antarmuka web dapat dilihat objek mana saja yang melakukan serangan dan objek yang aman. Untuk objek yang melakukan serangan maka diberi peringatan untuk ditindak lebih lanjut.

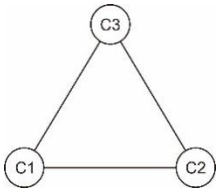
Topologi

Telah dibuat topologi jaringan yang dilakukan pada pelitian ini. Pada jaringan ini terdiri dari tiga buah ESP8266 sebagai objek yang dapat terhubung dengan server. Pembuatan server pada jaringan ini menggunakan Raspberrypi 3 b+ sebagai lokal server. Juga telah dibuatnya MySQL Database pada Raspberrypi sebagai penyimpanan nilai *trust* yang di dapat dari objek. Dapat dilihat topologi jaringan yang dibuat pada Gambar 2.

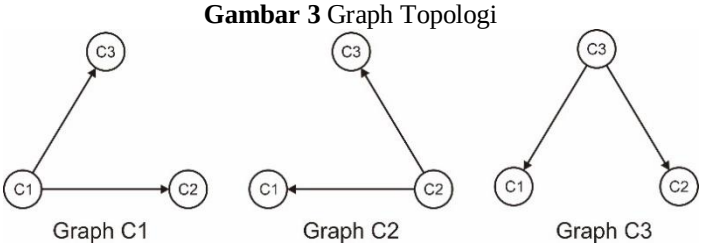


Gambar 2 Topologi Jaringan

Pada Gambar 2 dijelaskan bagaimana topologi yang dibuat. Selanjutnya topologi diatas diubah kedalam bentuk graph agar lebih mudah untuk melakukan perhitungan statistik matematika. Ini seperti yang sudah dilakukan pada penelitian [1],[10].



Gambar 3 Graph Topologi



Gambar 4 Graph Topologi Bagian

Pada Gambar 3 dijelaskan yaitu topologi antar objek yang terhubung dengan satu jaringan server yang sama diubah kedalam bentuk graph. Kemudian dipecah seperti pada Gambar 4, ini dimaksud untuk mempermudah proses perhitungan menjadi perbagian objek. Pada penelitian disini hanya membahas bagaimana proses untuk pedeteksian *On-off Attack*. Selebihnya telah dijelaskan pada paper [1], [10].

Pada proses pendeteksian disini menggunakan rumus matematika yang telah dibuat pada penelitian [1]. Dapat dilihat pada persamaan (1) dan (2).

$$nf = nb + ng$$
$$nf, nb, ng \in \mathbb{N} \quad (1)$$

$$nf = \begin{cases} berbahaya, & nf > 3 \\ tidak\ berbahaya, & nf \leq 3 \end{cases} \quad (2)$$

Keterangan:
nf = banyaknya semua serangan *trustbased attacks* dari sebuah objek
nb = banyaknya serangan *bad-mouthing attacks* dari sebuah objek
ng = banyaknya serangan *good-mouthing attacks* dari sebuah objek

Penjelasan dari rumus diatas yaitu. Pada proses pendeteksian, masing-masing objek melakukan penilaian terhadap objek lain yang selanjutnya diterima oleh server untuk diproses ketika setelah disimpan di database. Setiap objek diberi kesempatan untuk mengirimkan nilai selama lima periode, dengan masing-masing periode maksimal mengirimkan satu nilai dari setiap objek. Jika objek melakukan serangan *trustbased-attack* (*bad-mouthing*, *good-mouthing*) lebih dari tiga kali. Maka objek tersebut dianggap sebagai objek berbahaya yang melakukan serangan *on-off attack*.

Untuk melakukan pendeteksian dengan perhitungan matematika disini menggunakan pendekatan statistika, rumus yang sudah ditetapkan pada penelitian [1]. Dapat dilihat pada persamaan (3). Standar deviasi disini digunakan mencari nilai ambang batas atau *threshold*. Yang selanjutnya dipakai pada persamaan (4) untuk mengatahui batas nilai masing-masing dari *good-mouthing* dan *bad-mouthing*. Objek dikatakan melakukan *trustbased attack* jika pada saat proses pengecekan dengan perhitungan persaman (4) nilai trust sama dengan nilai *good-mouthing* dan *bad-mouthing* yang telah didapatkan pada pengoperasian (3). Untuk tahap pengecekaannya dimulai dari, proses perhitungan persamaan (3) dan (4) terlebih dahulu pada setiap periode. Lalu untuk tahap selanjutnya, *threshold* yang didapat dari periode sebelumnya digunakan untuk pengecekan *On-off Attack* pada periode setelahnya. Begitu seterusnya hingga periode terakhir.

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N}(Ti-\bar{T})^2}{N-1}} \quad (3)$$

$$G = \bar{T} + \sigma$$
$$B = \bar{T} - \sigma \quad (4)$$

Keterangan :
σ = standar deviasi
Ti = nilai *trust* dari objek ke-i
$\bar{T}$ = nilai *trust* rata-rata

N = banyaknya objek dalam jaringan
G = batas nilai *good-mouthing*
B = batas nilai *bad-mouthing*

Dari setiap objek yang terdeteksi melakukan serangan *On-off Attack* maka akan diberi tanda peringatan kepada pengguna admin server bahwa telah terjadi *On-off attack* pada objek ke-i. Pengguna dapat melihat hasil dari pendeteksian melalui web monitoring objek yang telah dibuat.

MySQL Database

Peneliti disini menggunakan MySQL Database sebagai database server yang difungsikan untuk penyimpanan data yang dikirim dari *objek* dan diterima oleh Raspberrypi selanjutnya dilakukan proses penyimpanan pada database tersebut. Peniliti menggunakan MySQL database karena selain kemudahan untuk menggunakannya juga mendukung pada penerapannya di Raspberrypi.

MQTT

Pada penelitian ini menggunakan protokol MQTT sebagai penghantar pesan melalui broker. Yang telah dirancang pada Raspberrypi menggunakan lokal address sebagai alamat broker. Untuk proses autentikasi disini, penulis menggunakan autentikasi yang sudah disediakan oleh paho MQTT. Mekanismenya yaitu dengan cara memberi username dan password pada broker. Jadi disetiap objek yang sudah terhubung pada jaringan, perlu melakukan autentikasi terlebih dahulu dengan cara mencocokkan username password yang dimiliki objek dengan yang dimiliki pada broker. Autentikasi disini bersifat satu arah saja.

PHP

Telah dibuat tampilan antarmuka sederhana untuk menampilkan hasil pendeteksian berupa tabel yang berisikan status dari setiap objek yang telah diperiksa nilai *trust* yang sudah dikirim ke database. Juga menampilkan berupa peringatan terhadap objek yang melakukan serangan *On-off Attack* untuk di tindak lanjut oleh admin server. Peneliti membuat keluaran berupa web bertujuan agar memudahkan admin server dalam melakukan pengawasan server dari jangkuan mana saja yang terhubung pada jaringan yang sama.

***Pseudocode* Algoritma**

Adapun *pseudocode* singkat dari algoritma yang telah dibuat untuk pendeteksian *On-off Attack* sebagai berikut.

Algorithm .. : Pseudocode for Detecting On-off Attack

```
Process Authentication
    If Objek <> autentikasi then
      Output "Objek tidak valid"
    Else
      nGood
      nBad
      goodM ← good-mouthing threshold
      badM ← good-mouthing threshold
      Get nilai_trust
      If badM < nilai_trust < goodM then
            Output "Nilai Trust Valid"
      Else if nilai_trust < badM then
            nBad += 1
            Output "Bad-mouthing Attack"
      Else
            nGood += 1
            Output "Good-mouthing Attack"
    On_off ← nBad + nGood
    If (on_off <> 0) & (on_off > 3) then
      Output "Objek melakukan On-off Attack"
```

Gambar 5 Algoritma Deteksi On-off Attack

Pada Gambar 5 menjelaskan alur algoritma yang dibuat dalam bentuk *pseudocode* yang diterapkan pada pada Raspberrypi untuk melakukan pengecekan objek. Berikut penjelasan singkat berdasarkan algoritma diatas.

1. Pertama setiap objek yang terhubung diperiksa terlebih dahulu valid atau tidak dengan proses autentikasi

2. Lalu jika objek valid maka selanjutnya objek diperiksa nilainya apakah melakukan *trustbased attack* atau tidak dengan cara membandingkan dengan nilai dari variabel goodM sebagai *good-mouthing threshold* dan badM sebagai *bad-mouthing attack threshold* yang sudah didapatkan.
3. Jika total nilai dari nGood dan nBad dijumlahkan dari lebih dari tiga kali serangan. Maka objek tersebut dikatakan melakukan serangan *On-off attack*.

Skenario pengujian

Telah dilakukan pengujian pendeteksian serangan *On-off Attack* pada penelitian ini. Pengujian dilakukan dengan mengasumsikan terlebih dahulu bahwa objek valid dan penyerang telah disimulasikan. Pada tahap pendeteksian serangan disini, yang pertama objek harus dipastikan dapat terautentikasi dengan MQTT Broker yang sudah digunakan. Jika ternyata bukan objek yang valid, maka objek tidak dapat mengirimkan nilai. Untuk objek yang telah terautentikasi selanjutnya dilakukan proses pengecekan nilai dari masing-masing objek. Dalam penelitian ini dilakukan sebanyak lima periode pengiriman nilai dari masing-masing objek. Penetapan lima periode ini bertujuan agar dapat diketahui objek mana saja yang melakukan *Trustbased Attack* seperti pada persamaan (2). Terdapat ketentuan dari setiap periode, untuk periode yang pertama nilai yang dikirimkan tidak dapat dikatakan melakukan *Trustbased Attack*. Nilai pada periode pertama ini digunakan hanya untuk menghitung standar deviasi dan batas dari *good-mouthing attack* dan *bad-mouthing attack* yang pertama. Yang selanjutnya nilai pada periode kedua dilakukan pengecekan *Trustbased attack* berdasarkan *threshold* yang didapat pada periode kedua. Selanjutnya pada periode kedua, dilakukan perhitungan standar deviasi dan batas dari *good-mouthing attack* dan *bad-mouthing attack*. Hasil perhitungan pada periode kedua ini digunakan untuk pengecekan *trusbased attack* pada periode ketiga. Untuk periode ketiga, keempat, dan kelima dilakukan proses yang sama untuk melakukan pengecekan *Trustbased Attack*. Setelah proses pengecekan *Trustbased Attack* selesai dilakukan, selanjutnya nilai *Trustbased Attack* dari setiap objek dijumlahkan. Objek dikatakan melakukan serangan *On-off Attack* jika melakukan serangan *Trustbased Attack* lebih dari tiga kali percobaan. Sebaliknya objek dikatakan valid jika melakukan serangan *Trustbased Attack* kurang dari tiga kali percobaan. Berikut ini merupakan tahapan skenario pengujian yang telah dilakukan.

1. Skenario pengujian eksekusi program aunthentikasi dan pengiriman nilai *trust* ke server
   Tujuan skenario pertama dilakukan pengujian pada tahap autentikasi dan pengiriman nilai *trust* ke server dan juga keberhasilan server untuk menyimpan pada database server. Pengujian dilakukan sebanyak lima kali percobaan. Kemudian dihitung seberapa besar persentase keberhasilan objek untuk melakukan autentikasi dan mengirim nilai trust ke server.
2. Skenario pengujian eksekusi program pendeteksian serangan *On-off Attack*
   Tujuan skenario kedua dilakukan pengujian tahap perhitungan statistik model untuk menentukan nilai standar deviasi, *bad-mouthing* dan *good-mouthing threshold* yang digunakan sebagai tahap pengecekan objek melakukan serangan *On-off Attack*. Pengujian dilakukan sebanyak lima kali percobaan. Dihitung seberapa besar presentase keberhasilan dalam mendeteksi objek yang melakukan serangan *On-off Attack*.
3. Skenario pengujian lama eksekusi program berjalan
   Tujuan skenario ketiga ini dilakukan pengujian lama waktu proses dari mulai program berjalan sampai program melakukan proses pengecekan *On-off Attack* selesai. Pengujian ini dilakukan sebanyak lima kali percobaan.

Berdasarkan skenario pengujian yang telah dibuat, pengujian dilakukan secara bertahap berdasarkan urutan program yang diproses terlebih dahulu. Untuk menghitung keberhasilan melakukan pendeteksian dan menghitung lama waktu eksekusi program dapat dilihat pada persamaan (5) dan (6).

$$Akurasi = \frac{Banyaknya\ keberhasilan}{Banyaknya\ pengujian}\ x\ 100\% \quad (5)$$

$$Total\ waktu\ (s) = Waktu\ Authentikasi + Waktu\ Pendeteksian \quad (6)$$

Seperti yang dijelaskan pada persamaan (5), untuk menghitung keberhasilan program yaitu banyaknya keberhasilan dalam lima kali pengujian dibagi dengan banyaknya pengujian yang dilakukan. Selanjutnya dikali 100% agar mengetahui dalam bentuk presentase keberhasilan.

Perhitungan lama total waktu yang berlangsung yaitu, lamanya waktu proses eksekusi program melakukan skenario pertama dijumlahkan dengan lamanya waktu proses eksekusi program melakukan skenario kedua. Semakin cepat total waktu proses eksekusi program, maka semakin cepat proses pedeteksian serangan.

4. Evaluasi

Tujuan pengujian pada penelitian dari tugas akhir ini adalah menerapkan implemntasi dari simulasi yang sudah dilakukan pada penelitian [1] dengan judul "The Detection of On-off Attack for the Internet of Things Objects". Hasil dari pengujian ini diharapkan bisa mendekati simulasi yang sudah lakukan.

4.1 Hasil Pengujian

Adapun tabel hasil dari pengujian setiap Skenario dibawah ini.

Tabel 1 Hasil pengujian skenario 1

Pengujian ke-	Skenario 1
1	100%
2	100%
3	100%
4	100%
5	100%

Tabel 2 Hasil pengujian skenario 2

Pengujian ke-	Skenario 2
1	100%
2	100%
3	100%
4	100%
5	100%

Tabel 3 Hasil pengujian skenario 3

Pengujian ke-	Skenario 3
1	0.524374 detik
2	0.587097 detik
3	0.563927 detik
4	0.540088 detik
5	0.543673 detik
Rata-rata	0.5518318 detik

4.2 Analisis Hasil Pengujian

Untuk menjelaskan hasil pengujian yang telah dilakukan, maka perlu analis agar mengetahui kesimpulan dari pengujian yang telah dilakukan. Berikut analisis berdasarkan pengujian skenario.

1. Skenario pertama

   Berdasarkan hasil pengujian pada Tabel 1 tingkat akurasi yang didapat yaitu 100% berdasarkan lima kali percobaan pengujian. Ini dikarenakan kemampuan objek untuk melakukan autentikasi dan mengirim nilai *trust* ke server sesuai dengan yang telah dibuat.
2. Skenario kedua

   Berdasarkan hasil pengujian pada Tabel 2 tingkat akurasi yang didapat dari setiap pengujian yaitu sebesar 100%. Tidak adanya kegagalan dalam semua percobaan, membuktikan proses pendeteksian berjalan sesuai harapan.
3. Skenario ketiga

   Hasil pengujian skenario ketiga ini didapatkan rata-rata waktu selama 0.5518318 detik. Berdasarkan dari setiap pengujian pada Tabel 3, program berjalan memakan waktu tidak lebih dari 0.5 detik, ini menunjukan banyaknya nilai *trust* yang dikirim bertambah pula waktu yang dibutuhkan untuk pendeteksian.

5. Kesimpulan

Kesimpulan berdasarkan hasil pengujian yang telah dilakukan, server dapat mendeteksi objek yang melakukan serangan *On-off Attack* dengan tingkat akurasi yang didapat sebesar 100% dan rata-rata waktu yang dibutuhkan untuk eksekusi program berjalan hingga berhasil mendeteksi serangan 0.5518318 detik.

Tentunya masih terdapat banyak kekurangan dan kelemahan dari penelitian ini. Maka penulis memberikan beberapa saran yaitu sebagai berikut.

1. Proses autentikasi masih berupa pada broker mqtt, akan lebih baik jika pada penelitian berikutnya dapat menggunakan autentikasi antar objek. Ini dimaksudkan untuk meningkatkan keamanan saat autentikasi berlangsung.
2. Dikarenakan keterbatasan kemampuan alat yang digunakan ESP8266 hanya dapat melakukan komputasi yang ringan. Sehingga server tidak dapat melakukan blok sebagai bentuk tindak langsung untuk objek yang melakukan serangan *On-off Attack*. Maka pada penelitian berikutnya akan lebih baik jika menggunakan alat yang lebih mampu untuk melakukan komputasi algoritma yang lebih kompleks.
3. Pengembangan antarmuka web admin dengan menambahkan fungsionaltas yang sesuai untuk melakukan monitoring objek *On-off Attack* seperti list perangkat, eksekusi perangkat yang melakukan serangan *On-off Attack* dan lainlain.
4. Pada penelitian berikutnya diharapkan pengujian dilakukan dengan cakupan alat yang lebih besar, agar dapat membuktikan pendeteksian *On-off Attack* menggunakan manajemen *Trustworthiness* dapat diterapkan lebih realistis.
5. Berdasarkan pengujian pada penelitian, metode ini masih terdapat kekurangan yaitu. Jika jumlah penyerangnya banyak, maka waktu untuk pemrosesan pun lebih lama dan ukuran database yang diperlukanpun lebih besar. Maka untuk penelitian berikutnya diperlukan sebuah metode untuk menangani hal ini.

Daftar Pustaka

[1] V. Suryani, S. Sulistyo, and Widyawan, "The Detection of On-Off Attacks for the Internet of Things Objects," *Proc. - 2018 Int. Conf. Control. Electron. Renew. Energy Commun. ICCEREC 2018*, pp. 1–5, 2018.

[2] Y. Chae, L. C. DiPippo, and Y. L. Sun, "Trust management for defending on-off attacks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 4, pp. 1178–1191, 2015.

[3] V. Suryani, S. Sulistyo, and W. Widyawan, "Simulation of trust-based attacks in Internet of Things," *MATEC Web Conf.*, vol. 154, p. 03014, 2018.

[4] S. M. Sony and S. B. Sasi, "On - Off attack management based on trust," *Proc. 2016 Online Int. Conf. Green Eng. Technol. IC-GET 2016*, 2017.

[5] J. Caminha, A. Perkusich, and M. Perkusich, "A smart middleware to detect on-off trust attacks in the Internet of Things," *2018 IEEE Int. Conf. Consum. Electron. ICCE 2018*, vol. 2018–Janua, pp. 1–2, 2018.

[6] C. V. L. Mendoza and J. H. Kleinschmidt, "Mitigating on-off attacks in the internet of things using a distributed trust management scheme," *Int. J. Distrib. Sens. Networks*, vol. 2015, 2015.

[7] X. Mao and J. McNair, "Effect of on/off misbehavior on overhearing based cooperation scheme for MANET," *Proc. - IEEE Mil. Commun. Conf. MILCOM*, pp. 1086–1091, 2010.

[8] J. H. Cho, A. Swami, and I. R. Chen, "A survey on trust management for mobile ad hoc networks," *IEEE Commun. Surv. Tutorials*, vol. 13, no. 4, pp. 562–583, 2011.

[9] F. Bao, I. R. Chen, and J. Guo, "Scalable, adaptive and survivable trust management for community of interest based internet of things systems," *Proc. - 2013 11th Int. Symp. Auton. Decentralized Syst. ISADS 2013*, 2013.

[10] V. Suryani, S. Sulistyo, and W. Widyawan, "Two-phase security protection for the Internet of Things object," *J. Inf. Process. Syst.*, vol. 14, no. 6, pp. 1431–1437, 2018.