NETWORK FORENSICS UNTUK MENDETEKSI SERANGAN *FLOODING* PADA PERANGKAT *INTERNET OF THINGS (IoT)*

RANDI RIZAL

14917156

Tesis diajukan sebagai syarat untuk meraih gelar Magister Komputer

Konsentrasi Forensika Digital

MAGISTER TEKNIK INFORMATIKA

PROGRAM PASCASARJANA FAKULTAS TEKNOLOGI INDUSTRI

UNIVERSITAS ISLAM INDONESIA

2018

Lembar Pengesahan Pembimbing

NETWORK FORENSICS UNTUK MENDETEKSI SERANGAN *FLOODING* PADA PERANGKAT *INTERNET OF THINGS* (IoT)

RANDI RIZAL
14917156

Yogyakarta, Juni 2018

Pembimbing,

Dr. Imam Riadi, M.Kom.

Lembar Pengesahan Penguji

***NETWORK FORENSICS* UNTUK MENDETEKSI SERANGAN *FLOODING* PADA PERANGKAT *INTERNET OF THINGS* (IoT)**

RANDI RIZAL

14917156

Yogyakarta, Juni 2018

Tim Penguji,

Dr. Imam Riadi, M.Kom
Ketua _____

Yudi Prayudi, M.Kom
Anggota I _____

Dr. Bambang Sugiantoro, MT
Anggota II _____

Mengetahui,

Ketua Program Pascasarjana Fakultas Teknologi Industri

Universitas Islam Indonesia

Dr. R. Teduh Dirgahayu, ST.,M.Sc.

Abstrak

NETWORK FORENSICS UNTUK MENDETEKSI SERANGAN *FLOODING* PADA PERANGKAT *INTERNET OF THINGS* (IoT)

Hari ini adalah era *Internet of Things* (IoT), jutaan mesin atau perangkat seperti *smart city*, detektor asap, jam tangan, pelacakan mobil otomatis, pendeteksian *smartphone*, *smart lighting*, pemantauan suhu dll sedang terhubung ke Internet. Di internet ada berbagai perangkat yang saling berhubungan dengan perangkat lain yang berbagi teknologi yang berbeda dan standar yang berbeda. Munculnya teknologi baru di berbagai bidang itu juga menimbulkan tantangan baru di bidang investigasi forensik. Karena akan ada banyak tantangan baru bagi penyelidik forensik. Alat-alat terbaru dan aliran proses yang dilakukan tidak akan memenuhi infrastruktur *Internet of Things* (IoT) yang sangat terdistribusi pada saat ini. Peneliti forensik akan memiliki banyak tantangan untuk menghadapi pengumpulan bukti dari komponen yang terinfeksi pada perangkat *Internet of Things* (IoT) dan juga akan menghadapi kesulitan untuk menganalisis bukti-bukti tersebut.

Network forensics untuk mendeteksi serangan *flooding* pada perangkat *Internet of Things* (IoT) menganalisis suatu file log dengan mencatat aktivitas yang dianggap berbahaya. *File log* yang merupakan barisan data untuk menyimpan informasi mengenai segala tindakan, kejadian dan aktifitas yang terjadi di dalam sebuah sistem jaringan. Selanjutnya digunakan untuk proses investigasi analisis forensik jaringan dengan menggunakan metode *Generic Network Forensics Process Model* yang merupakan ilmu digital forensik yang berkaitan dengan tahap-tahap untuk menemukan sumber serangan dan mendapatkan bukti-bukti serangan yang bersumber dari file log.

Hasil penelitian yang telah dilakukan telah mampu mendeteksi serangan *flooding* pada perangkat *Internet of Things* (IoT) dengan menggunakan bantuan *tools* aplikasi *network analyzer wireshark* untuk mengcapture *traffic log* dalam sistem jaringan. Sejumlah 10 IP *address* yang melakukan tindakan illegal ke dalam perangkat *Internet of Things* (IoT). Hasil analisis penelitian serangan *flooding* dapat menemukan barang bukti digital hasil investigasi forensik jaringan. Berdasarkan hasil pengujian tersebut dapat dinyatakan hasil sudah sesuai dengan tujuan yang diharapkan, sehingga dapat disimpulkan penelitian ini berhasil berjalan dengan baik dan lancar.

***Kata Kunci** : IoT, Network Forensics, Flooding Attack, Digital Investigations.*

Abstract

NETWORK FORENSICS FOR DETECTING FLOODING ATTACK ON INTERNET OF THINGS (IoT) DEVICE

Today is the era of the Internet of Things (IoT), millions of machines or devices such as smart city, smoke detectors, watches, automatic car tracking, smartphone detection, smart lighting, temperature monitoring etc. are being connected to the Internet. In the internet of things, there are various devices which are interconnected to the other devices which share different techniques and the different standards. The rise of new technology in various fields it also makes rise to the new challenges in the area of the forensic investigation. As there will be many new challenges to the forensic investigators. The recent tools and the process flow carried out will not meet the highly distributed and current infrastructure of the IoT. A Forensic researcher will have a lot of challenges to face in collecting the piece of evidence from the infected component on the IoT device and also will face complication to analyze those evidence.

Network forensics to detect flooding attacks on the Internet of Things (IoT) device analyzes a log file by recording activities that are considered harmful. Log files that are data rows to store information about any actions, events and activities that occur within a network system. It is then used for the investigation of forensic network analysis using the Generic Network Forensics Process Model method which is a digital forensic science that deals with the stages to find the source of the attack and obtain evidence of attacks originating from the log file.

The results of the research have been able to detect flooding attacks on the device of Internet of Things (IoT) by using tools wireshark network analyzer application to capture traffic logs in the network system. There are 10 IP addresses that illegally commit to the Internet of Things (IoT) device. The results of the research analysis of flooding attacks can find digital evidence of network forensic investigation results. Based on the test results can be declared the results are in accordance with the expected objectives, so it can be concluded this research successfully run well and smoothly.

Keywords : *IoT, Network Forensics, Flooding Attack, Digital Investigations.*

Pernyataan Keaslian Tulisan

Dengan ini saya menyatakan bahwa tesis ini merupakan tulisan asli dari penulis, dan tidak berisi material yang telah diterbitkan sebelumnya atau tulisan dari penulis lain terkecuali referensi atas material tersebut telah disebutkan dalam tesis. Apabila ada kontribusi dari penulis lain dalam tesis ini, maka penulis lain tersebut secara eksplisit telah disebutkan dalam tesis ini.

Dengan ini saya juga menyatakan bahwa segala kontribusi dari pihak lain terhadap tesis ini, termasuk bantuan analisis statistik, desain survei, analisis data, prosedur teknis yang bersifat signifikan, dan segala bentuk aktivitas penelitian yang dipergunakan atau dilaporkan dalam tesis ini telah secara eksplisit disebutkan dalam tesis ini.

Segala bentuk hak cipta yang terdapat dalam material dokumen tesis ini berada dalam kepemilikan pemilik hak cipta masing-masing. Apabila dibutuhkan, penulis juga telah mendapatkan izin dari pemilik hak cipta untuk menggunakan ulang materialnya dalam tesis ini.

Yogyakarta, Juni 2018

Randi Rizal, ST

Daftar Publikasi

Rizal, R., Riadi, I., and Prayudi,Y. (2018). Network Forensics for Detection Flooding Attack on Internet of Things (IoT) Device. *International Journal of Cyber Security and Digital Forensics (IJCSDF)*.

<u>Publikasi yang menjadi bagian dari tesis</u>

Sitasi publikasi 1

Kontributor	Jenis Kontribusi
Randi Rizal, ST.	Mendesain eksperimen (70%) Menulis *paper* (70%)
Dr. Imam Riadi, M.Kom.	Mendesain eksperimen (20%) Menulis dan mengedit *paper* (15%)
Yudi Prayudi, M.Kom.	Mendesain eksperimen (10%) Menulis dan mengedit *paper* (15%)

Halaman Kontribusi

“ **Tidak ada kontribusi dari pihak lain** ”.

Halaman Persembahan

Karya ini saya persembahkan kepada :

1. Dua orang pahlawan dalam hidupku, ayahanda ***Usep Saepudin*** dan (almh) ibunda ***Entin Patimah***. Kasih sayang dan cintanya tidak mungkin terbalas. Semoga Allah SWT membalasnya dengan menempatkannya di surga-Nya kelak. Aminnn...
2. Istriku tercinta ***Neni Nuraeni*** yang tidak hentinya memberi motivasi dan dukungannya dengan penuh kesabaran. Kesayangan dan jagoan abii dan ummie ***Zhafran Ibra Rizal*** yang menjadi penghibur hati dan inspirasi.
3. Kakakku Ranti Marantika Dewi dan Keluarga, adik-adikku Nisa Maulida dan Rizky Ridwan yang telah memberikan motivasi kepada penulis. Tetaplah semangat dalam hidup berkeluarga dan teruslah selalu termotivasi dalam kebaikan.
4. Keluarga besar pondok pesantren Al-Furqon yang telah memberikan banyak pengalaman dalam memaknai arti kehidupan.
5. Semoga kita selalu berada dalam kasih sayang dan lindungan Allah SWT. Amiinn

Kata Pengantar

Puji syukur kehadirat Allah SWT atas segala limpahan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan tesis dengan judul : ***"Network Forensics untuk mendeteksi serangan Flooding pada perangkat Internet of Things (IoT)"***, semoga karya ini bermanfaat bagi penulis, instansi, masyarakat dan ilmu pengetahuan itu sendiri khususnya bidang digital forensik.

Sebagai ungkapan rasa hormat dan terimkasih atas dukungan, saran, bimbingan dan jasa-jasa yang membantu penulis dalam menyelesaikan karya ini, penulis menyampaikan ucapan terima kasih yang sedalam – dalamnya kepada :

1. Dr. R. Teduh Dirgahayu, ST., M.Sc. selaku Direktur Program Pascasarjana Fakultas Teknologi Industri.
2. Dr. Imam Riadi, M.Kom. selaku dosen pembimbing satu, yang telah banyak membantu penulis dalam memberikan ide, saran dan kritiknya.
3. Yudi Prayudi, M.Kom. selaku dosen pembimbing dua, yang telah banyak memberikan kemudahan baik pengarahan maupun bimbingan selama pengajuan tesis.
4. Dr. Bambang Sugiantoro selaku dewan penguji tesis yang telah memberikan banyak masukan dalam penyusunan tesis ini.
5. Dosen – dosen program studi Magister Teknik Informatika yang telah memberikan bekal ilmu pengetahuan kepada penulis, semoga ilmunya menjadi amal *jariyah* di dunia maupun di akhirat.
6. Rekan – rekan mahasiswa program studi Magister Teknik Informatika, khususnya kosentrasi forensika digital angkatan X (sepuluh).
7. Pusat Studi Forensika Digital (PUSFID) dan Staff IT Centrum.
8. Semua Staff di MTI-UII yang telah banyak membantu penulis.

Akhirnya penulis menyadari bahwa hasil karya ini masih banyak kekurangan dan kelemahan, untuk itu saran dan kritik yang konstruktif akan sangat membantu agar tesis ini dapat menjadi lebih baik.

Yogyakarta, Juni 2018

Randi Rizal

Daftar Isi

Lembar Pengesahan Pembimbing i
Lembar Pengesahan Penguji ii
Abstrak iii
Abstract iv
Pernyataan Keaslian Tulisan v
Daftar Publikasi vi
Halaman Kontribusi vii
Halaman Persembahan viii
Kata Pengantar iix
Daftar Isi x
Daftar Tabel xii
Daftar Gambar xiii
BAB 1 Pendahuluan 1
1.1 Pendahuluan 1
1.2 Rumusan Masalah 3
1.3 Batasan Masalah 3
1.4 Tujuan Penelitian 3
1.5 Manfaat Penelitian 3
1.6 Metodologi Penelitian 4
1.7 *Rivew* Penelitian 4
1.8 Sistematika Penulisan 9
BAB 2 Kajian Pustaka 10
2.1 Forensik Digital 10
2.2 Forensik Jaringan 11
2.3 Forensik *Internet of Things* (IoT) 13
2.2.1 Arsitektur Forensik IoT 15
2.2.2 IoT *Environment* 17
2.4 Keamanan *Internet of Things* (IoT) 19
2.2.1 Serangan pada Perangkat IoT 22
2.5 Arduino UNO 24
2.2.1 Sejarah 25

2.2.2 Hardware 25
2.2.2 Software 26
2.6 Bluetooth HC-05 27
BAB 3 Metodologi Penelitian 30
3.1 *Literatur Review* 30
3.2 Identifikasi Sistem 31
3.3 Konfigurasi Arduino Bluetooth Hc-05 31
3.4 Simulasi Kasus 33
3.5 Analisis Investigasi Forensik 35
3.6 Laporan 36
BAB 4 Pembahasan 38
4.1 *Literatur Review* 38
4.2 Identifikasi Sistem 39
4.3 Simulasi Serangan *Flooding* 40
4.4 Analisis dan Investigasi Forensik 40
4.4.1 Implementasi Forensik Jaringan 41
4.4.2 Analisis Model Proses Forensik 41
BAB 5 Kesimpulan dan Saran 48
5.1 Kesimpulan 48
5.2 Saran 48
Daftar Pustaka 49
LAMPIRAN A 51

Daftar Tabel

Tabel 1.1 *Literatur Review* 7
Tabel 1.2 *Literatur Review* 8
Tabel 2.1 *Digital Forensics Process Models* 14
Tabel 2.2 Perbandingan Forensik Traditional dan IoT 18
Tabel 3.1 Pengelompokan Data 35
Tabel 3.2 Diagram Grafik Serangan 35
Tabel 4.1 Klasifikasi Paket Serangan 47

Daftar Gambar

Gambar 1.1 Perangkat IoT yang terhubung Tahun 2012-2020. 2
Gambar 1.2 Metodologi Penelitian. 4
Gambar 2.1 *Forensic Science* 10
Gambar 2.2 Turunan Ilmu Forensik 11
Gambar 2.3 Pembagian sub cabang *network forensics* 12
Gambar 2.4 *Network Forensics Process Model* 12
Gambar 2.5 Arsitektur IoT 16
Gambar 2.6 *IoT Forensics* 17
Gambar 2.7 Serangan terhadap Perangkat IoT 22
Gambar 2.8 *DDoS Attacks* 23
Gambar 2.9 Blok Diagram *Arduino Board* 24
Gambar 2.10 *Hardware Arduino Uno* 25
Gambar 2.11 Bentuk fisik Bluetooth Hc-05 28
Gambar 3.1 Alur Metodologi Penelitian 30
Gambar 3.2 Metode *Generic Framework for Network Forensics* 31
Gambar 3.3 Identifikasi sistem perangkat IoT 31
Gambar 3.4 Skema Konfigurasi Arduino Bluetooth Hc-05 32
Gambar 3.5 Simulasi sistem serangan *Flooding* pada perangkat IoT 33
Gambar 3.6 Tahap simulasi serangan *Flooding* 34
Gambar 3.7 Metode *Generic Framework for Network Forensics* 35
Gambar 3.8 Laporan 35
Gambar 4.1 *Pairing* Perangkat Bluetooth 38
Gambar 4.2 Arsitektur Forensik Jaringan 39
Gambar 4.3 *Test Ping IP victim* 39
Gambar 4.4 Serangan *Flooding* dengan *LOIC* 40
Gambar 4.5 Arsitektur Forensik Jaringan *IoT Device* 41
Gambar 4.6 *Detection of Findings Log* 42
Gambar 4.7 *Data Collection Stage* 43
Gambar 4.8 *Hash Evidence of Log Traffic* 43
Gambar 4.9 *IO Graph Traffic Log* 44
Gambar 4.10 *Traffic Log in Wireshark* 44
Gambar 4.11 Filter ip.src 45

Gambar 4.12 Follow UDP Stream 45
Gambar 4.13 Hasil *follow* UDP..... 46
Gambar 4.14 Hasil *Capture Frame*..... 46

Bab 1 Pendahuluan

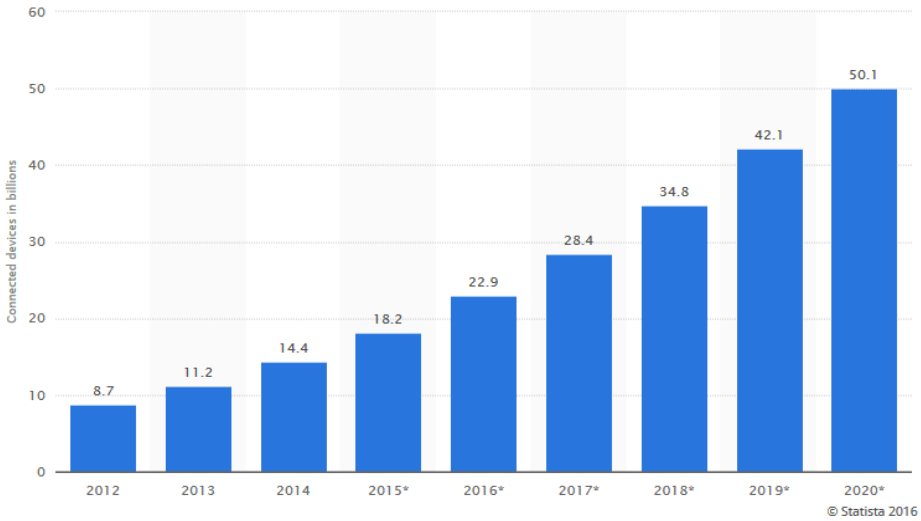
1.1 Pendahuluan

Kemajuan internet dan pengembangan inovatif dari perangkat elektronik pintar mengarah kepada pengembangan dari sebuah prototype komputer baru yaitu IoT *(Internet of Things)*. IoT dianggap menjadi investasi masa depan internet yang bekerja pada komunikasi *Machine to Machine* (M2M) dan *Radio Frequency Identification* (RFID) (Zawoad, et al. 2015). Tujuan utama dari IoT *(Internet of Things)* adalah untuk memungkinkan pertukaran data yang aman antara perangkat dunia nyata dan aplikasi.

Internet of Things (IoT) telah menjadi cukup akrab dalam beberapa tahun terakhir. Banyaknya perangkat rutinitas sehari-hari semakin terhubung dengan Internet yang mencakup banyak kemampuan seperti penginderaan, otonomi dan kesadaran kontekstual. Perangkat IoT termasuk komputer pribadi, laptop, smartphone, tablet dan perangkat rumah yang tertanam lainnya (Hossain, et al. 2015). Perangkat ini terhubung satu dengan yang lain dan berbagi jaringan yang sama untuk saling berkomunikasi. Semua perangkat ini terhubung dengan sensor tertentu untuk menditeksi kondisi sekitarnya dan bekerja sesuai dengan situasi. Perangkat ini juga diprogram untuk mengambil sebuah keputusan secara otomatis atau menginformasikan sesuai dengan pengguna sehingga pengguna dapat membuat keputusan terbaik.

Jaringan interkoneksi pada *Internet of Things (IoT)* dapat menghubungkan berbagai objek yang memiliki identitas pengenal serta alamat IP, sehingga dapat saling berkomunkasi dan bertukar informasi data. Objek-objek dalam IoT dapat menggunakan maupun menghasilkan layanan-layanan dan saling bekerjasama untuk mencapai suatu tujuan bersama, sehingga dapat membawa banyak kemajuan di bidang teknologi aplikasi dan layanan yang dapat membawa manfaat bagi pengembangan bisnis global. Banyak perangkat yang langsung terhubung ke internet untuk saling berbagi informasi dan komunikasi.

Menurut laporan analisis statistika tahun 2016, seperti yang terihat pada gambar 1.1 akan ada sekitar 51 miliar perangkat pada tahun 2020 yang terhubung ke internet dengan media konektivitas yang berbeda. Di Internet terdapat berbagai perangkat yang saling berhubungan dengan perangkat lain dengan berbagi teknologi yang berbeda dan standar yang berbeda.



Gambar 1.1 Perangkat IoT yang terhubung
Tahun 2012 – 2020

Karena banyaknya jumlah perangkat yang terhubung ke *Internet of Things* (IoT) pada akhirnya mendapat perhatian serangan *hacker* dalam melanggar mekanisme keamanan (Hossain, dkk, 2015). Untuk menyelidiki serangan tersebut kita perlu menerapkan aspek forensik digital di *Internet of Things* (IoT) yang bisa kita sebut dengan istilah IoT Forensik (Zawoad, dkk, 2015). Salah satu isu yang masih menjadi kelemahan dalam pengimplementasian *Internet of Things* (IoT) adalah masalah keamanan dan privasi. Serangan terhadap keamanan *Internet of Things* (IoT) dapat mencakup serangan terhadap label *Radio Frequency Identification* (RFID), jaringan komunikasi maupun pada privasi data (Ernita Dewi Meutia, 2015).

Serangan *Flooding* menjadi salah satu bentuk serangan *Denial of Service* (DoS) terhadap jaringan perangkat *Internet of Things* (IoT) yang mengakibatkan suatu sistem akan terbanjiri oleh data – data secara terus menerus dalam waktu singkat dan juga mengakibatkan lalu lintas jaringan menjadi sangat padat.

Dalam forensika digital diperlukan suatu model proses forensik jaringan untuk mengintegrasikan persyaratan kesiapan forensik kedalam rancangan sistem. Framework *Generic Network Forensics* merupakan sebuah model forensik yang dirancang untuk menggabungkan keamanan dengan mempunyai fungsi kemampuan forensik digital yang lebih akurat dalam setiap tahapnya. Mengumpulkan banyak fase yang tersedia dalam model forensik digital tetapi dibangun pada fase-fase yang khusus untuk forensik jaringan. (E.S. Pilli, et al, 2010).

Dalam penelitian ini implementasi framework *Generic Network Forensics* bertujuan melakukan investigasi forensik jaringan untuk mendeteksi serangan *flooding* pada perangkat *Internet of Things* (IoT).

1.2 Rumusan Masalah

Dari latar belakang diatas dapat ditarik permasalahan untuk dijadikan perumusan masalah antara lain :

1. Bagaimanakah menguji framework *Generic Network Forensics Process Model* untuk menemukan serangan *flooding* dari perangkat *Internet of Things* (IoT)?
2. Bagaimanakah menguji kinerja untuk mendeteksi serangan *flooding* pada perangkat *Internet of Things* (IoT)?

1.3 Batasan Masalah

Batasan-batasan masalah yang digunakan dalam penelitian ini adalah sebagai berikut :

1. Penelitian dilakukan untuk pencarian serangan *flooding* pada perangkat *Internet of Things* (IoT) yang terinfeksi.
2. Fokus penelitian ini hanya dilakukan proses forensik jaringan pada perangkat *Internet of Things* (IoT).
3. *Device* yang digunakan adalah Arduino UNO Bluetooth Hc 05
4. Mendeteksi trafik logs file

1.4 Tujuan Penelitian

Berdasarkan rumusan yang dibuat maka dapat diambil tujuan dari penelitian ini. Tujuan dari dibuatnya penelitian ini antara lain :

1. Menguji framework *Generic Network Forensics Process Model* untuk mendeteksi serangan flooding dari perangkat *Internet of Things (IoT)*.
2. Melakukan pengujian untuk mendeteksi kinerja serangan *flooding* pada perangkat *Internet of Things (IoT).*

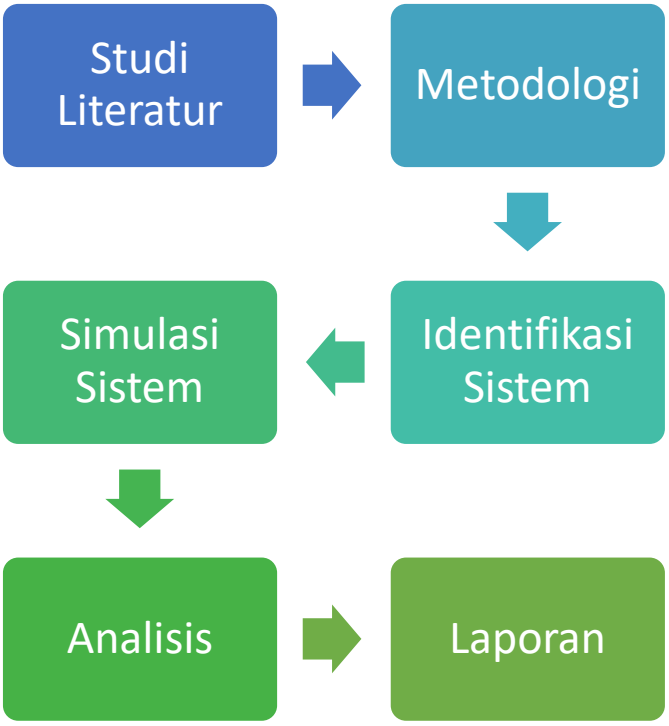
1.5 Manfaat Penelitian

Manfaat yang dihasilkan dari penelitian ini antara lain :

1. Memberi kontribusi terhadap pengembangan ilmu *Digital Forensics* khususnya tentang *Internet of Things* (*IoT) Forensic*s *Device*, dengan memanfaatkan framework *Generic Network Forensic Model* untuk menenukan serangan *flooding* pada perangkat *Internet of Things* (IoT).
2. Mengetahui trafik dan log file pada perangkat *Internet of Things* (IoT) Arduino UNO dan Bluetooth HC 05.
3. Mengetahui proses forensik jaringan yang terjadi dalam *Digital Forensics Investigation* pada perangkat *Internet Of Things* (*IoT)*.

1.6 Metodologi Penelitian

Perlu disusun langkah-langkah penyelesaian penelitian secara sistematis yang disebut sebagai metodologi penelitian. Penelitian ini menggunakan beberapa tahapan metode penelitian yaitu :



Gambar 1.2. Metodologi Penelitian

Gambar 1.2 menunjukkan penelitian ini menggunakan 6 tahapan metodologi penelitian yakni (1) Studi *Literatur* (2) Metodologi (3) Identifikasi Sistem (4) Simulasi Sistem (5) Analisis (6) Laporan.

1.7 Review Penelitian

Edewede Oriwoh, et al. (2013) menjelaskan tentang IoT dengan mengusulkan pertama penerapan pendekatan zona 1-2-3 ke (IoT) *Internet of Things* terkait dengan investigasi Forensika Digital (FD). Kedua, memperkenalkan model Next-Best-Thing Triage (NBT) untuk digunakan dalam hubungannya dengan Zona 1-2-3 mendekati mana yang diperlukan dan sebaliknya. Kedua pendekatan ini sangat penting untuk proses forensik digital dari perspektif IoT *(Internet of Things)* yang bersifat khas sebagai sumber bukti IoT, lingkungan IoT secara keseluruhan dan atribut lainnya yang unik. Kombinasi dari atribut – atribut ini menentukan perlunya pendekatan forensik digital secara sistematis untuk suatu kejadian. Dua pendekatan yang diusulkan dirancang sebagai pilar untuk melayani dalam merespon suatu kejadian, meningkatkan efisiensi dan afektivitas penyelidikan terkait IoT *(Internet of Things)* dengan memaksimalkan penggunaan waktu yang tersedia dan memastikan identifikasi dan akuisisi bukti yang relevan.

Yuanjun Song (2013) membahas tentang perancangan aplikasi untuk platform arsitektur keamanan IoT terutama terdiri dari model pengalamatan dan model akses kontrol. Pertama, penelitian ini merancang model keamanan dalam pengalamatan IoT *(Internet of Things)* untuk melindungi informasi rahasia pribadi dari objek. Kedua, merancang model akses keamanan objek IoT *(Internet of Things)* baru untuk melindungi komunikasi antara pemohon akses dan objek. Tujuannya bisa tercapai apabila persyaratan diatas terpenuhi sehingga menghasilkan untuk melindungi kerahasiaan, integritas dan privasi dalam proses objek IoT *(Internet of Things)* dalam menangani dan mengakses dengan biaya yang relatif rendah.

Qi Jing, Athanasios V.Vasilakos,.dkk. (2014) menjelaskan tentang masalah keamanan diantara semua tantangan lainnya. Sebagaimana IoT dibangun atas dasar internet, masalah keamanan internet juga akan muncul di IoT *(Internet of Things)*. IoT *(Internet of Things)* mengandung tiga lapisan yaitu lapisan persepsi, lapisan transportasi dan lapisan aplikasi, yang mana penelitian ini menganalisa masalah keamanan setiap lapisan secara terpisah dan mencoba untuk menemukan solusi dari masalah baru. Penelitian ini juga menganalisis masalah lintas *layer* dalam integrasi secara heterogen dan masalah keamanan secara rinci dan membahas isu-isu keamanan IoT *(Internet of Things)* secara keseluruhan dan mencoba untuk mencari solusi. Secara keseluruhan inti penelitian ini adalah membandingkan masalah keamanan antara IoT *(Internet of Things)* dan jaringan tradisional serta membahas keamanan dari IoT.

Robert Hegarty, David Lamb, Andrew Attwood (2014) menjelaskan tentang teknik baru IoT *(Internet of Things)* yang diperlukan untuk mengatasi tantangan dan memanfaatkan arsitektur dengan proses yang digunakan untuk forensik digital dalam rangka mendapatkan akses ke sumber bukti yang potensial. Menetapkan secara menyeluruh tantangan IoT dan mengidentifikasi area kunci sebagai solusi untuk proses forensik digital.

Shams Zawoad dan Ragib Hasan (2015) menjelaskan dalam mengatasi forensik IoT *(Internet of Things)* dari dua perspektif yaitu pertama melaksanakan prosedur forensik digital di infrastruktur IoT, ketika IoT *(Internet of Things)* menjadi target serangan atau digunakan untuk melancarkan serangan. Kedua menentukan fakta baru / tidak diketahui dengan memanfaatkan infrastruktur IoT *(Internet of Things)*. Berdasarkan karakteristik IoT *(Internet of Things)* dan prosedur forensik digital dalam mengidentifikasi tantangan proses forensic digital, dimana bukti tetap berada dalam infrastruktur IoT. FAIoT adalah model forensik yang menjadi infrastruktur IoT *(Internet of Things)* untuk mendukung kedua perspektif.

IoT dalam penelitian Ernita Dewi Meutia (2015) menjelaskan tentang 3 hal yaitu: manajemen identitas, otentikasi dan otorisasi yang dapat menjamin keamanan dan privasi dari sistem IoT. Akan tetapi untuk menjaga integritas data pada RFID, sensor maupun basis data dari serangan tampering (mengubah atau memodifkasi data), maka data harus disimpan dalam bentuk

terenkripsi. Banyak metode enkripsi yang telah dikembangkan, antara lain dengan menggunakan hash key dan algoritma AES.

Dalam penelitian Ardimulya I dan Imam Riadi (2015) memberi gambaran tentang Algoritma *Density K-Means* yang digunakan untuk membantu melakukan pengelompokan *Log database* didasarkan pada tingkat bahaya rendah, sedang atau tinggi.Akan tetapi Algoritma *Density K-Means* hanya mampu membagi data ke dalam 3 kelompok. Aplikasi forensik menggunakan *triage framework* dapat diterapkan sebagai identifikasi awal untuk mengakui keberadaan serangan *Denial of Service* (DoS), berdasarkan atas tahapan yaitu klasifikasi perangkat elektronik, melakukan ekstraksi perangkat digital, melakukan ekstraksi fitur sebagai masukan dalam proses analisis, algoritma pengolahan dalam analisis yang digunakan dan pelaporan temuan yang berfungsi sebagai panduan untuk mendukung keberadaan insiden utuk segera mengambil langkah – langkah tindakan perbaikan dini untuk mengurangi tingkat kerusakan pada sistem.

Meet Tilva dan Vandana Rohokale (2016) memberikan gambaran tentang model penggunaan forensik jaringan untuk mendeteksi sebuah paket berbahaya yang terinfeksi dan mengidentifikasi sumber paket yang berbahaya pada lingkungan simulasi. Mensimulasikan serangan Denial of Service (DoS) untuk menganalisa perangkat IoT *(Internet of Things)* yang terinfeksi. Memantau dan komunikasi paket pada *contiki coja simulator* untuk mengidentifikasi forensik jaringan. Peneliti mengusulkan penelitian ini dapat diperluas untuk membuat investigasi forensik jaringan yang lebih aman dengan mekanisme berbasi Log untuk membuat sistem lebih aman dan menjadi mudah untuk menemukan kerusakan tertentu pada sistem.

Nurul Huda Nik Zulkipli, Ahmed Alenezi dan Gary B. Wills (2017) menjelaskan tentang forensik IoT dalam menjembatani tantangan dalam digital forensika dengan *Internet of Things* (IoT). Untuk memenuhi karakteristik forensik IoT, ada dua pendekatan dalam penyelidikan forensik IoT yang diusulkan dengan menekankan pada fase pra-penyelidikan dan menerapkan penyelidikan *real time* untuk memastikan data dan bukti potensial yang dikumpulkan dan disimpan selama penyelidikan.

Dalam penelitian Tanveer Zia, Peng Liu, Weili Han (2017) menjelaskan tantangan terbesar forensik IoT adalah sebuah perangkat yang heterogen dalam sistem IoT dan kurangnya standar yang terpadu. Dalam penelitian ini penyelidikan forensik digital dari perspektif IoT, berpendapat bahwa selain praktik forensik digital tradisional, penting untuk memiliki forensik khusus aplikasi untuk memastikan pengumpulan bukti dalam konteks aplikasi IoT tertentu dengan mempertimbangkan tiga aplikasi IoT dan memperkenalkan model yang berhubungan dengan bukan hanya forensik tradisional tetapi berlaku dalam digital serta proses forensik khususnya aplikasi. Memperkenalkan model investigasi forensik digital spesifik aplikasi dengan menunjukkan jenis artefak yang akan menjadi forensik yang penting dalam forensik IoT.

Tabel 1.1. *Literatur Review*

No.	Peneliti / Tahun	Metode dan Pendeteksian	Jenis Serangan Tertinggi			Metode Investigasi	Kesimpulan
			Malware	Brute Force	Flooding / DoS		
1	Oriwoh, Edewede, et al. (2013)	Model Next-Best-Thing Triage (NBT) untuk forensika digital	–	–	–	Model Proses Forensik	Meningkatkan efesiensi dan efektifitas proses forensik perangkat IoT.
2	Yuanjun Song (2013)	Aplikasi untuk platform arsitektur keamanan IoT	√	–	–	Model Pengalamatan dan Akses Kontrol	Melindungi kerahasian, integritas dan privasi dalam proses objek IoT
3	QiJing, Athanasios V.Vasilakos, et al (2014)	Perspektif, tantangan investigasi pada keamanan IoT	–	–	–	–	Membandingkan keamanan IoT dan jaringan tradisional.
4	Robert Hegarty, David Lamb, Andrew A (2014)	Tantangan dan pendekatan bukti digital pada IoT	–	–	–	Model Proses Forensik	Memanfaatkan arsitektur forensik digital untuk mendapatkan akses ke sumber bukti.
5	Shams Zawoad, Ragib Hasan (2015)	Model Forensik-Aware IoT (FAIoT). Membangun kesadaran Forensik	–	–	–	Model Proses Forensik	Konseptual Forensik-Aware IoT (FAIoT) untuk investigasi forensik yang terpercaya di lingkungan IoT
6	Ernita Dewi Meutia (2015)	Metode enkripsi dengan hash key dan algoritma AES pada sistem IoT	–	√	–	–	Menghasilkan tiga hal untuk privasi dan keamanan IoT yaitu: manajemen identitas, otentikasi dan otorisasi.
7	Ardymulya I, Imam Riadi (2015)	Aplikasi triage framework dan Metode Algoritma *Density K-Means*	–	–	√	–	Pengelompokan *Log database* dengan tingkat bahaya rendah, sedang dan tinggi

Tabel 1.2. *Literatur Review*

No.	Peneliti / Tahun	Metode dan Pendeteksian	Jenis Serangan Tertinggi			Metode Investigasi	Kesimpulan
			Malware	Brute Force	Flooding / DoS		
8	Meet Tilva dan Vandana Rohokale (2016)	Forensika jaringan untuk mendeteksi paket yang berbahaya pada IoT	–	–	√	Model Forensik jaringan pada IoT	Mendeteksi sebuah paket berbahaya dan mengidentifikasi sumber paket pada lingkungan simulasi contiki coja.
9	Nurul Huda, Ahmed A, dan Gary B.W (2017)	Menjembatani tantangan dalam forensik digital dan IoT	–	–	–	Metode Forensik jaringan pada IoT	Fase pra-penyelidikan dan menerapkan penyelidikan *real time* untuk data dan bukti potensial.
10	Tanveer Zia, Peng Liu, Weili Han (2017)	Tinjauan tantangan forensik digital pada IoT	–	–	–	Model forensik khusus aplikasi pengumpulan bukti konteks aplikasi IoT	Memperkenalkan model aplikasi spesifik investigasi forensik digital jenis artefak dalam forensik IoT.
Usulan Penelitian		***Network Forensics* untuk mendeteksi serangan *flooding* pada perangkat IoT**	√	–	√	Model Proses Forensik Jaringan	Mengasilkan traffic log file perangkat IoT dan mengidentifikasi level jenis serangan berbasis log
		Dalam penelitian ini akan dilakukan simulasi serangan flooding pada perangkat IoT secara langsung, tidak menggunakan simulator contiki coja. Tujuan akhirnya adalah mendapatkan traffic log file yang melewati perangkat IoT yang ditangkap atau direkam menggunakan *tools network analyzer wireshark*, kemudian dianalisa sehingga terdeteksi paket mana yang terinfeksi oleh serangan dan mendapatkan paket IP yang menjadi sumber serangan. Maka model proses investigasi forensik akan membantu dalam memvisualisasikan *file log* serangan agar dapat dipahami oleh masyarakat awam yang tidak faham dengan digital forensik atau menjadi bukti legal secara hukum dalam persidangan.					

1.8 Sistematika Penulisan

Untuk mempermudah proses pembahasan dalam penelitian yang dibuat, maka dibuat sistematika penulisan pada penelitian ini :

BAB I PENDAHULUAN

Pendahuluan merupakan pengantar terhadap permasalahan yang akan dibahas. Didalamnya menguraikan tentang gambaran suatu penelitian yang terdiri dari latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian, serta sistematika penulisan.

BAB II KAJIAN TEORI

Pada Bab ini menjelaskan teori-teori yang digunakan untuk memecahkan masalah dalam penelitian ini. Teori yang dibahas pada bagian ini merupakan teori yang berhubungan *digital forensic, network forensics model, flooding attack* dan *Forensic in Internet of Things (IoT) environment.*

BAB III METODOLOGI PENELITIAN

Bab ini membahas tentang langlah-langkah penelitian dan gambaran umum langkah penyelesaian. Berisi tentang arsitektur forensik *Internet of Things* (IoT), tantangan forensik jaringan pada perangkat IoT dan pengujian model framework *Generic Network Forensik Process Model*.

BAB IV PEMBAHASAN

Hasil dan Pembahasan, berisi tentang pembahasan penyelsaian masalah yang diangkat, pengujian kinerja dan penentuan hasil analisis.

BAB V KESIMPULAN DAN SARAN

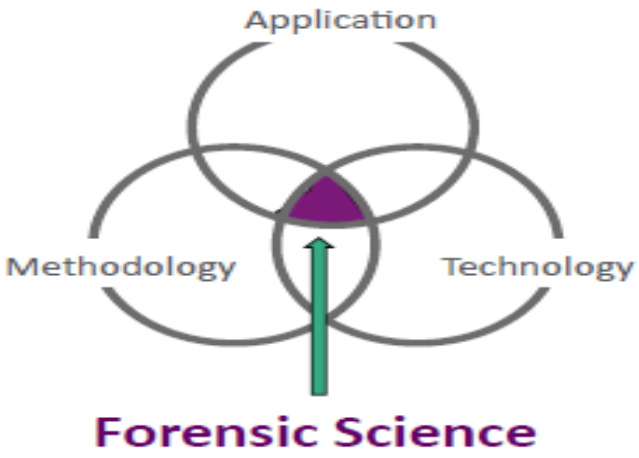
Simpulan dan Saran, memuat kesimpulan-kesimpulan dari hasil penelitian dan saran-saran yang perlu diperhatikan berdasar keterbatasan yang ditemukan dan asumsi-asumsi yang dibuat selama melakukan penelitian dan juga rekomendasi yang dibuat untuk pengembangan penelitian selanjutnya.

Bab 2 Kajian Pustaka

2.1 Forensik Digital

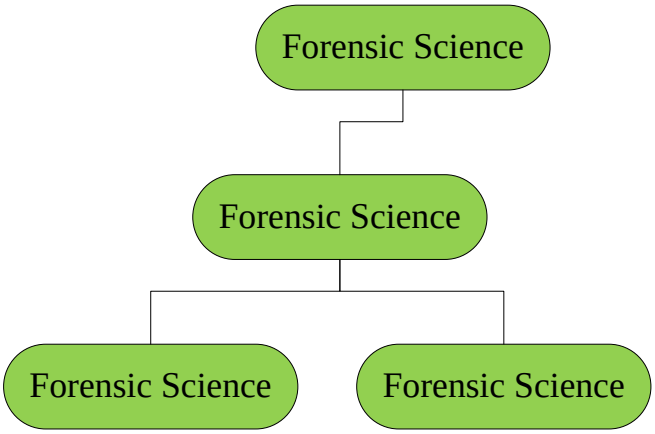
Forensik digital (*digital forensics*) merupakan cabang ilmu dan metode yang digunakan di dalam pemulihan, pengumpulan, validasi, identifikasi, analisis, interpretasi, dokumentasi, dan presentasi barang bukti digital yang diperoleh dari sumber digital dengan tujuan untuk memfasilitasi atau membuat kemajuan dalam proses rekonstruksi kejadian kriminal, atau membantu dalam antisipasi tindakan yang mengganggu jalannya investigasi yang telah direncanakan (Palmer, 2001). Dari definisi tersebut dapat diketahui bahwa forensik digital (*digital forensics)* berguna dalam proses investigasi suatu tindak kejahatan kriminal yang melibatkan adanya barang bukti elektronik atau digital.

Menurut K Franke (2010), ilmu forensik merupakan metodologi aplikasi yang benar dari *spectrum* yang luas dari disiplin ilmu untuk menjawab pertanyaan yang signifikan terhadap sistem hukum. Seperti yang terlihat dari gambar 2.1 menunjukan bahwa metode forensik terdiri dari pendekatan untuk melakukan tugas seperti: (1) Menyelidiki tempat kejadian perkara, (2) Mengumpulkan jenis data-data dan menganalisis jenis bukti yang telah ditemukan, (3) Mengidentifikasi, mengelompokkan, mengukur, memisahkan antara pelaku, objek dan proses penanganan, (4) Membangun hubungan, mengasosiasi, dan merekontruksi ulang kejadian, (5) Menggunakan barang bukti di pengadilan.



Gambar 2.1 Forensic Science

Forensic science ini mempunyai beberapa cabang turunan ilmu forensik yang dikembangkan menjadi digital forensik, *computer* forensik dan *network* forensik. Berikut gambar yang menjelaskan turunan ilmu forensik :



Gambar 2.2 Turunan Ilmu Forensik

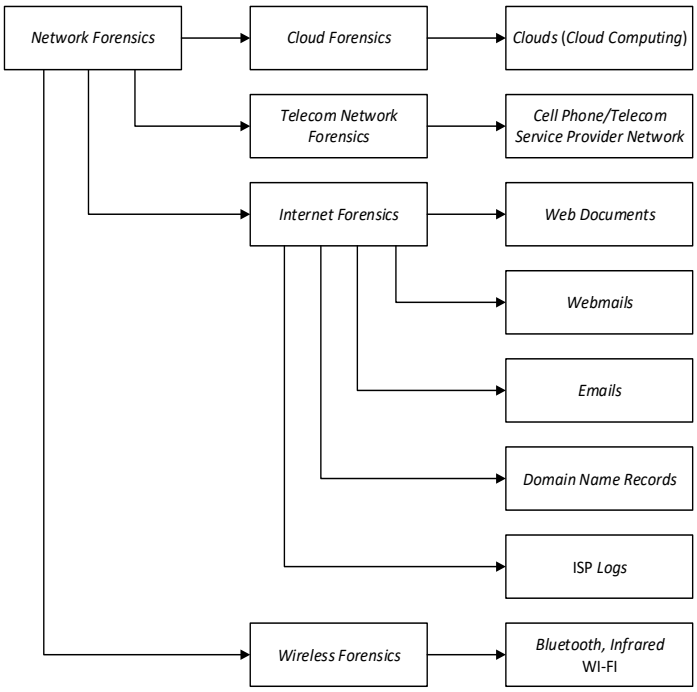
Penelitian T.Charles dan M. Pollock (2015) menyebutkan bahwa digital forensic merupakan metode ilmiah untuk melestarikan, mengoleksi, validasi, identifikasi, analisis, interpretasi, dokumentasi, dan presentasi digital untuk tujuan menfasilitasi atau merekontruksi peristiwa ditemukan tindak kriminal, atau membatu untuk mengantisipasi tindakan yang tidak sah atau terbukti mengganggu proses perencanaan operasi. (Charles & Pollock, 2015)

Penelitian Aleksandar V, Heis S, Mellisa I (2014) mendefinisikan *digital forensic* sebagai penggunaan ilmiah yang diturunkan dengan bukti metode identifikasi, pengumpulan, transportasi, penyimpanan, analisis, diartikan, dipresentasikan dan didistribusikan kembali dari bukti digital yang berasal dari sumbernya. Dengan mendapatkan otorisasi untuk semua kegiatan yang berinteraksi langsung dengan penyelidikan, melestarikan barang bukti, melacak barang bukti atau melakukan rekontruksi peristiwa ditemukannya insiden.

2.2 Forensik Jaringan

Salah satu dari enam cabang utama dalam ilmu *digital forensics* adalah forensik jaringan (*Network Forensics)*. *Network forensics* atau disebut sebagai forensik jaringan adalah cabang *digital forensics* yang menggunakan teknik yang terbukti secara ilmiah untuk mengumpulkan, menggunakan, mengidentifikasi, menguji, menghubungkan, menganalisis, dan mendokumentasikan barang bukti digital dari beberapa sumber digital yang aktif memproses dan mengirimkannya. Hal ini bertujuan untuk menyingkap fakta, mengukur keberhasilan suatu aktivitas yang tidak terotorisasi, seperti mengganggu, merusak, atau menyusup ke dalam komponen sistem, dan juga untuk menyediakan informasi yang berguna dalam pemulihan sistem terkait dari aktivitas merugikan tersebut (Palmer, 2001).

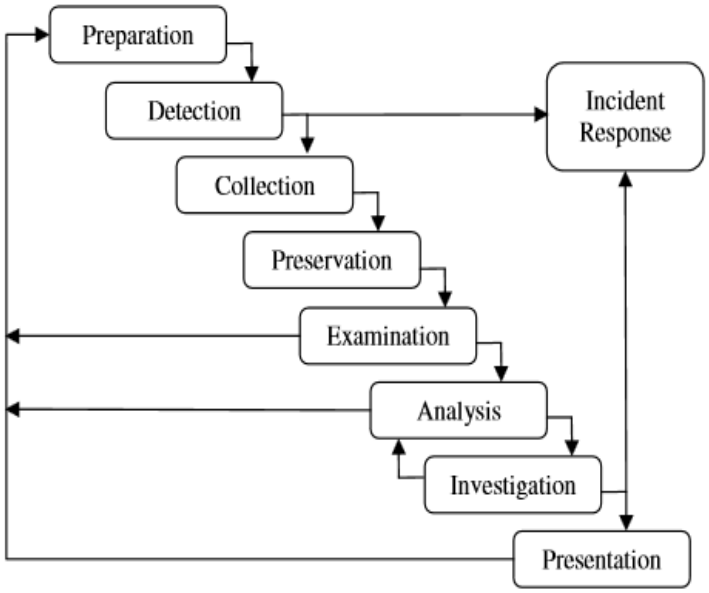
Network forensics erat kaitannya dengan informasi yang dinamis dan rentan. Maksudnya adalah informasi yang diperlukan dalam *network forensics* mudah hilang setelah ditransmisikan melalui jaringan. Selain itu, barang bukti dalam *network forensics* seperti file-file log di dalam *host* yang telah disusupi juga dapat dihapus oleh penyerang (Karie & Venter, 2014).



Gambar 2.3 Pembagian sub cabang *network forensics*

Sumber : Karie & Venter, 2014

Dalam makalah tentang *"A Survey About Network Forensics"* penulis mengusulkan model investigasi forensik jaringan. Model yang diusulkan terdiri dari beberapa tahap proses penyelidikan forensik jaringan. Gambar 2.4 menunjukan model dari forensik jaringan yang digambarkan memiliki sembilan tahapan (E.S. Pilli, R.C. Joshi, & R. Niyogi, 2010).



Gambar 2.4 Network Forensic Process Model

- **Tahap Persiapan** : Tujuan utamanya adalah untuk mendapatkan otorisasi yang diperlukan dan jaminan secara hukum.
- **Tahap Penemuan** : Menghasilkan peringatan atau peringatan yang menunjukkan pelanggaran keamanan.
- **Tahap Penanganan Insiden** : Berlaku hanya ketika penyelidikan dimulai selama serangan itu berlangsung.
- **Tahap Pengumpulan** : Bagian paling sulit karena data mengalir dengan cepat dan tidak mungkin menghasilkan jejak dengan hal yang sama.
- **Tahap Pemeliharaan** : Bukti asli tetap aman bersamaan dengan penghitungan hash.
- **Tahap Pemeriksaan** : Memeriksa tahap sebelumnya. Semua data yang disembunyikan atau diubah adalah untuk pengungkapan yang dilakukan oleh penyerang.
- **Tahap Analisis** : Bukti yang dikumpulkan dianalisis untuk menemukan sumber gangguan atau serangan.
- **Tahap Investigasi** : Penggunaan informasi yang dikumpulkan dalam tahap analisis dan fokus untuk menemukan penyerang.
- **Presentasi** : Tahap akhir untuk memproses model. Berikut dokumentasi dibuat dan laporan yang dihasilkan dan ditampilkan kepada pihak atau otoritas yang lebih tinggi.

2.3 Forensik *Internet of Things* (IoT)

Lembaga penegak hukum, organisasi swasta bahkan individu yang akrab dengan kombinasi barang bukti baik elektronik ataupun digital dihasilkan dari proses investigasi forensik. Seperti perangkat elektronik yang menjadi tren saat ini tampaknya akan terus berlanjut. Sebagai contoh bagi polisi bisa untuk memeriksa sebuah operator jaringan selular, apakah *driver* menggunakan telepon selulerketika menyelidiki insiden lalu lintas jalan (GM Casualty Reduction Partnership 2014) (Haines, 2007).

Perangkat *Internet of Things* (IoT) meningkatkan lingkungan data digital saat ini dengan data pribadi yang berpotensi signifikan dan pengaturan konteks pemasukan data. Kami mengidentifikasi persyaratan untuk investigasi di *Internet of Things (IoT)* dengan terlebih dahulu mempertimbangkan pekerjaan dari McKemmish (McKemmish, 1999) yang mendefinisikan Digital Forensik sebagai proses identifikasi, melindungi, menganalisis dan menyajikan bukti digitaldengan cara yang dapat diterima secara hukum. Penerapan definisi ini untuk penyelidikan di IoT menimbulkan sejumlahtantangan.Untuk mengembangkan pemahaman tentang tantangan inidalam proses forensik digitalmempertimbangkan dua model utama untuk investigasi forensik digital.

Tabel 2.1 memberikan gambaran dari model proses digital forensik yang diusulkan oleh McKemmish (McKemmish, 1999) dan NIST (Kent, Chevalier, Grance, Dang, & Kent, 2006).

Stage	McKemmish	Kent
1.	***Identification,*** dalam tahap ini lokasi dan format bukti diidentifikasi untuk memungkinkan suatu mekanisme yang tepat yang akan ditentukan untuk tujuan memulihkan bukti. Bukti digital dapat ditemukan di komputer, ponsel, smart card, dll	***Collection,*** meliputi identifikasi, pemeliharaan dan akuisisi barang bukti yang relevan
2.	***Preservation,*** sangat penting bahwa bukti yang dipelihara seperti dalam banyak kasus itu akan menjadi subyek pengawasan yudisial. Dalam beberapa keadaan perubahan data yang tidak dapat dihindari. Dalam kasus ini perubahan harus diminimalkan dan proses yang menyebabkan perubahan didokumentasikan bersama dengan penjelasannya.	***Examination,*** menggunakan alat otomatis dan manual untuk mengekstrak data yang penting.
3	***Analysis,*** terdiri dari ekstraksi, pengolahan dan interpretasi bukti digital. Ini merupakan unsur utama dari komputasi forensik. Berikut ekstraksi, pengolahan seringkali diperlukan untuk membuat data dibaca manusia. Pengolahan data yang diambil dapat menjadi bagian dari tahap ekstraksi atau tahap terpisah dalam dirinya sendiri	***Analysis,*** turunan dari informasi yang berguna dari hasil tahap pemeriksaan.
4	***Presentation,*** tahap akhir dari proses yang melibatkan presentasi dari kedua bukti dan proses dimana bukti itu dikumpulkan bersama dengan kualifikasi penyaji.	***Reporting,*** berkaitan dengan penyusunan dan penyajian bukti dan proses analisis forensik.

Tabel 2.1 : Digital Forensics Process Models

Identifikasi dan pemeliharaan bukti dalam penyelidikan forensik digital di lingkungan yang muncul selalu disajikan sebuah tantangan. Taylor et al (Taylor, Haggerty, Gresty, & Hegarty, 2010) mengemukakan bahwa bukti standar yang dinilai dalam penyelidikan forensik digital mungkin harus diubah untuk mengakomodasi perubahan sifat bukti digital dari lingkungan *Cloud Computing*dan untuk penyelidikan *Internet Of Things (IoT)*.

Oriwoh et al (Oriwoh, Jazani, Epiphaniou& Sant, 2013) Identifikasipemeliharaan sebagai tantangan dan menyarankan bahwa dalam menjalanipemeriksaanperangkattidak harus dimatikan untuk pemeliharaan modifikasi, pembuatan dan waktu mengakses sebuah file. Pernyataan mereka kemungkinan diambil dari investigasi forensik digital konvensional. Namun, situasinya jauh lebih kompleks dalam penyelidikan *Internet Of Things (IoT)*. Berfikir harus diberikan kepada sumber daya terbatas yang tersedia di perangkat, meninggalkan perangkat yang sedang berjalan di tempat kejadian adalah suatu kesalahan dan yang lebih penting dapat mengakibatkan penimpaan data yang tersimpan karena kemampuan penyimpanan yang dibatasi. Oleh karena itu, pertimbangan diperlukan untuk menentukan apakah perangkat harus dimatikan atau dalam keadaan berjalan.

Ekstraksi dan pemeliharaan data dari perangkat dan layanan yang berjalan di *Internet of Things* (IoT) akan menyajikan tantangan. Kepemilikan format data, protokol dan antar muka secara fisik akan mempersulit proses ektraksi barang bukti (Miorandi, Sicari, De Pellegrini & Chlamtac, 2012).

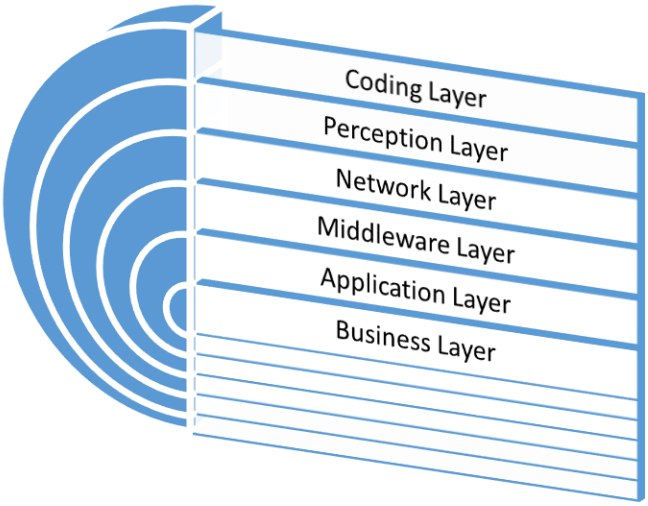
Beberapa skema mendistribusikan informasi ke node yang berdekatan dengan topologi yang sama atau layanan cloud eksternal. Dalam scenario ini, peneliti harus dapat mengidentifikasi manfaat untuk penyelidikan dalam penggalian data dari node lain, BTS atau layanan cloud (Attwood, Merabti & Abuelmaatti, 2011). Pendekatan ini bisa menjadi layak dan dapat mengatasi beberapa tantangan yang berkaitan dengan penggalian data dari perangkat dengan penyimpanan terbatas. Data yang disimpan dan diproses di IoT dapat bersifat sensitif. Kami mengandaikan bahwa cara dimana perusahaan – perusahaan besar bergabung dan proses data mungkin bisa menjadi subjek penyelidikan forensik digital masa depan.

Oriwoh et al (2013) mengidentifikasi tantangan yang ditimbulkan oleh perangkat yang melintasi batas – batas wilayah hukum, sementara kita setuju ini adalah sebuah tantangan. Hal ini sangat mungkin bahwa dalam pengiriman data antara perangkat IoT dan platform komputasi *cloud* yang didistribusikan secara global melintasi batas ini secara jauh dan lebih sering. Sehingga muncul pertanyaan apakah penekanan proses investigasi forensik harus pada perangkat atau data, dan apakah perangkat dapat dilihat sebagai aspek metadata dari data atau sebaliknya.

2.3.1 IoT Forensics Architecture

Lebih dari 25 miliar benda yang diharapkan akan terhubung pada tahun 2020 yang merupakan jumlah yang sangat besar sehingga menggunakan arsitektur internet dengan protokol TCP/IP, yang diadopsi pada tahun 1980 tidak bisa menangani jaringan sebesar *Internet of Things* (IoT) yang menyebabkan kebutuhan untuk mengatasi berbagai masalah keamanan dan *Quality of Service (QoS)* serta bisa mendukung aplikasi jaringan yang ada menggunakan protocol terbuka (Jian An, dkk. 2012). Tanpa jaminan privasi yang tepat, *Internet of Things* (IoT) tidak mungkin diadopsi oleh banyak kebutuhan. Oleh karena itu perlindungan data dan privasi pengguna merupakan tantangan utama bagi *Internet of Things* (IoT).

Untuk pengembangan lebiih lanjut dari IoT, sejumlah multi arsitektur keamanan berlapis yang diusulkan. Dalam penelitian MiaoWu dkk. (2010) mengusulkan lima arsitektur berlapis menggunakan fitur terbaik dari arsitektur jaringan manajemen telekomunikasi internet dan berdasarkan masing – masing TCP/IP dan model TMN. Demikian pula enam arsitektur berlapis juga diusulkan berdasarkan hirarkis jaringan (XuCheng, 2012).



Gambar 2.5 Arsitektur IoT

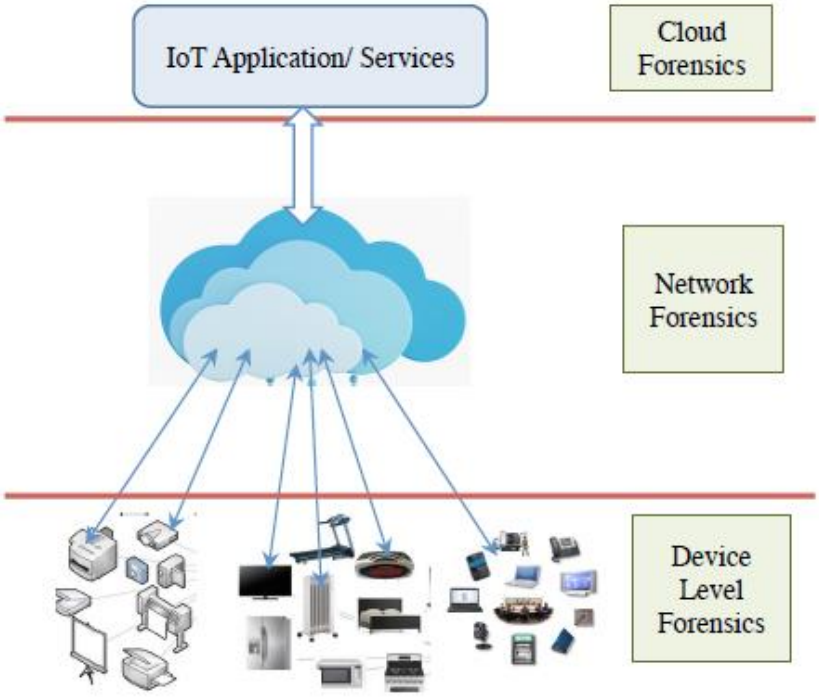
Jadi secara umum arsitektur IoT itu dibagi menjadi enam lapisan seperti yang ditunjukan pada gambar 2.3. Keenam lapisan IoT tersebut dijelaskan sebagai berikut :

- Coding Layer : Lapisan pengkodean adalah dasar dari IOT yang menyediakan identifikasi terhadap suatu obyek yang menarik. Dalam lapisan ini, setiap objek diberi ID unik yang membuatnya mudah untuk membedakan objek – objek tersebut (Xu Cheng, dkk. 2012).
- Perception Layer : Ini adalah lapisan perangkat dari IoT yang memberikan arti fisik untuk setiap objek. Terdiri dari sensor data yang berbentuk acuh tak acuh seperti label RFID, sensor IR atau jaringan sensor lainnya yang bisa merasakan suhu, kelembaban, kecepatan dan lokasi dari objek (DebasisBandyopadhyay, dkk. 2012). Lapisan ini mengumpulkan informasi yang berguna dari sensor perangkat objek yang terhubung dengan mereka dan mengubah informasi menjadi sinyal digital yang kemudian diteruskan ke lapisan jaringan untuk tindakan lebih lanjut.
- Network Layer : Tujuan dari lapisan ini adalah menerima informasi yang berguna dalam bentuk sinyal digital dari *Perception Layer* dan ditransfer ke sistem pengolahan di Middleware Layer melalui media transmisi seperti WiFi, Bluetooth, WiMax, Zigbee, GSM, 3G dengan protokol seperti IPv4, IPv6, MQTT, DDS dan lain – lain. (Ying Zhang, 2011).
- Middlware Layer : Lapisan ini memproses informasi yang diterima dari perangkat sensor. Ini termasuk teknologi seperti Cloud Computing, komputasi di mana – mana yang menjamin akses langsung ke database untuk menyimpan semua informasi yang diperlukan di dalamnya. Mengunakan industri *intelegent processing*, informasi tersebut secara otomatis diproses dan ditindak berdasarkan hasil pengolahan informasi.

- Application Layer : Lapisan ini menyadari sebuah aplikasi IoT untuk semua jenis industri, berdasarkan data yang diproses. Karena aplikasi ini mempromosikan pengembangan IoT sehingga sangat membantu dalam pengembangan skala besar dalam jaringan IoT (Maio Wu, 2010). Aplikasi IoT yang berhubungan bisa menjadi rumah pintar, transportasi cerdas, planet pintar dan lain – lain.
- Business Layer : Lapisan ini mengelola aplikasi dan layanan dari IoT dan bertanggung jawab untuk semua penelitian yang berkaitan dengan IoT. Aplikasi ini menghasilkan model bisnis yang berbeda untuk strategi bisnis yang lebih efektif (Rafiullah Khan, dkk. 2012).

2.3.2 IoT Forensics Environment

Internet Of Things (IoT) forensics merupakan salah satu cabang khusus dalam forensik digital dimana semua tahapan penawaran dibahas dengan infrastruktur IOT untuk menemukan fakta – fakta tentang kejahatan yang terjadi di lingkungan IoT. Forensik IoT dilakukan dalam tiga tingkat forensik yaitu tingkat forensik cloud, forensik tingkat jaringan, dan forensic tingkat perangkat seperti dijelaskan pada gambar 2.3 (Sundresan Perumal, et al. 2015).



Gambar 2.6 *IoT Forensics*

Sumber : Sundresan, Norita, & Valliappan 2015

- ***Forensik tingkat perangkat :*** Di sini, di tingkat ini penyidik telah mengumpulkan bukti potongan kecil dari perangkat yang bekerja pada *Internet of Things (IoT)*. Perangkat yang terinfeksi diidentifikasi dan dilaporkan.

- ***Forensik tingkat jaringan :*** Dalam tingkat ini serangan yang berbeda diidentifikasi melalui jaringan log dan menditeksi paket berbahaya di jaringan yang berbeda, bisa menjadi jaringan *Local Area Network (LAN)* atau *Wide Area Network (WAN).*
- ***Forensik tingkat Cloud :*** Ini adalah salah satu bagian terpenting dalam *Internet of Things* (IoT) karena semua data yang dihasilkan dari perangkat IoT dan jaringan IoT disimpan di cloud. Sebagai cloud menawarkan banyak fitur seperti penyimpanan yang besar dan infrastuktur dengan banyak ruang.

Maria Karyda dan Lilian Mitrou (2007) menerangkan bahwa bagaimana lingkungan IoT forensik bekerja dan tiga tingkat forensik perlu dilakukan dalam skenario IoT untuk mengetahui sumber sebenarnya dari perangkat yang terinfeksi atau pelanggaran jaringan. Penelitian Edewede Oriwoh, dkk (2013) menjelaskan dibagian ini akan dilakukan perbandingan parameter bagaimana system bekerja sebenarnya dan bagaimana solusi yang diusulkan untuk suatu permasalahan.

Parameters	**Perbandingan Forensik Traditional dan Forensik IoT**	
	Forensik Traditional	**Forensik IoT**
Bukti	Komputer, cloud, perangkat, server, gateways, *mobile devices*.	*Home appliances, car tags, readers, embedded system, nodes.*
Perangkat yang terhubung	Miliaran perangkat yang tersambung	50 miliar perangkat terhubung pada 2020 menurut Gartner.
Jaringan	Wired, Wireless, Bluetooth Wireless network, Internet	RIFD, Sensor Network
Protocols	Ethernet wireless (802.11 a,b,g,n), Bluetooth, Ipv4 and Ipv6	RFID, Rime
Ukuran barang Bukti Digital	Hingga terabyte data	Hingga Exabyte data

Table 2.2 Perbandingan Forensik Traditional and IoT

Alat dan teknologi tradisional tidak dirancang sepenuhnya untuk melakukan forensik di lingkungan *Internet of Things* (IoT) karena menghadapi banyaknya tantangan [Buric J,.dkk,2015]. Di bagian ini kami akan mengidentifikasi tantangan yang dihadapi untuk penyelidikan forensik di lingkungan *Internet of Things* (IoT) [Zawoads, 2015].

I. Identifikasi perangkat yang dikompromikan pada IoT.

Penjahat itu. Misalnya, ada sejumlah perangkat di kampus dan jika ada perangkat yang disusupi dan mengalami pelanggaran pada jaringan dan mengekstrak beberapa file pribadi, akan sangat sulit untuk menemukan sumber perangkat yang terinfeksi. Tantangan ini seperti menemukan jarum di tumpukan jerami.

II. Mengumpulkan dan menganalisis data.

Setelah identifikasi, datanglah analisis dan pengumpulan yang merupakan tugas yang cukup sulit untuk menemukan bukti. Fase ini merupakan fase yang sangat penting dan bergantung pada fase lain yang juga menghasilkan kesalahan ke fase lainnya.

III. Organisasi Data

Berbagai macam data yang dihasilkan oleh perangkat IoT membuat fase pengumpulan dan analisis menantang. Log yang tepat harus diatur untuk menghindari kerumitan data dan file.

IV. Pemeliharaan Barang Bukti

Langkah terakhir dari penyelidikan forensik adalah bahwa pemeriksa forensik menyajikan informasi yang dikumpulkan dan bukti di depan pengadilan hukum. Seperti dalam perbandingan presentasi bukti forensik tradisional adalah mudah daripada forensik pada lingkungan *Internet of Things* (IoT) karena itu tugas yang menantang karena anggota juri tidak memiliki pengetahuan yang cukup dibandingkan dengan orang teknis. Mereka juga merasa kompleks untuk dipahami.

2.4 Keamanan *Internet of Things* (IoT)

Salah satu tantangan yang harus diatasi untuk mendorong implementasi IoT secara luas adalah factor keamanan. IoT merupakan sebuah sistem yang majemuk. Kemajemukannya bukan hanya karena keterlibatan berbagai entitas seperti data, mesin, RFID, sensor dan lain-lain, tetapi juga karena melibatkan berbagai peralatan dengan kemampuan komunikasi dan pengolahan data. Banyaknya entitas dan data yang terlibat, membuat IoT menghadapi resiko keamanan yang dapat mengancam dan membahayakan konsumen. Ancaman ini utamanya dilakukan dengan cara memungkinkan orang yang tidak berhak untuk mengkases data dan menyalah gunakan informasi personal, memfasilitasi serangan terhadap sistem yang lain, serta mengancam keselamatan personal penggunanya.

Ancaman-ancaman yang dapat mempengaruhi entitas IoT sangat beragam, tergantung dari target serangantersebut. Roman dkk dalam [5] mengkategorikan ancaman terhadap IoT sebagai berikut:

1. Denial of Service, serangan menyebabkan pihak yang sah tidak dapat mengakses layanan.
2. Merusak secara fsik objek-objek dalam IoT.
3. Eavesdropping; serangan pasif yang dapat dilakukanpada berbagai kanal komunikasi dengan tujuan mengekstrak data dari aliran informasi.

4. Node capture; penyerang mengekstrak informasi dari node maupun dari infrastruktur lain yang memilikikemampuan penyimpanan data.
5. Controlling; di mana penyerang berusaha mendapatkankontrol terhadap entitas IoT dan mengganggu layanan maupun data dari entitas tersebut.

Berbagai jenis ancaman di atas, dapat menyerang berbagai entitas dalam IoT, terutama RFID dan jaringan sensor [3].

A. Label RFID.

Secara alami, RFID rentan terhadap ancaman keamanan dan privasi. Sifatnya yang tidak memerlukan kontak langsung dan berkomunikasi secara nirkabel dengan memanfaatkan gelombang elektromagnetik, menyebabkan interaksi dengan label RFID dapat dilakukan tanpa kontak fisik sehingga mudah diserang tanpa terdeteksi. Ancaman terhadap keamanan RFID dapat terjadi baik pada label, pembaca, host maupun pada kanal komunikasinya. Label RFID sebagai sarana pengenal objek, dibuat dengan biaya yang rendah. Mengingat harganya yang murah, sulit melengkapi label RFID dengan mekanisme enkripsi dan pemrograman yang kuat. Akibatnya label RFID rentan terhadap serangan seperti pencurian, penggandaan maupun pemodifkasian data.

Pada sisi kanal komunikasi, RFID yang berkomunikasi dengan memanfaatkan gelombang elektromagnetik, rentan terhadap interferensi. Interferensi akan mengakibatkan kesalahan data dalam proses komunikasi antara label dan pembaca. Dengan mengirimkan sinyal penginterferensi, penyerang dapat menghambat link komunikasi, sehingga pembaca tidak dapat membaca data yang benar, dan menyebabkan serangan denial of service dan data tampering.

Selain itu, pembaca RFID juga dapat dipalsukan sehingga komunikasi antara pembaca dan host dapat diserang dengan mudah. Penyerang dapat melakukan serangan middleman (pembaca lain yang diletakkan ditengah jalur komunikasi dan berperan seolah-olah sebagai pembaca yang sesungguhnya), *eaves dropping* maupun menginterferensi pertukaran data secara langsung maupun tidak langsung antara pembaca dengan host. Akibatnya, label tidak dapat diidentifkasi dengan benar, atau terjadi kesalahan identifkasi.

B. Wireless Sensor Network (WSN)

WSN merupakan teknologi kunci yang memungkinkan terwujudnya IoT. Dengan WSN jaringan dan layanan dapat diintegrasikan menjadi infrastruktur IoT. WSN dan jaringan komunikasi yang dimanfaatkan pada IoT bekerja secara nirkabel, sehingga mudah diserang dan diinterferensi.

Prinsip pengamanan informasi pada WSN dan jaringan komunikasi mengikuti prinsip *confdentiality, integrity* dan *availability*, berdasarkan prinsip ini, serangan yang dapat mengancam WSN, dapat dikategorikan dalam 3 kategori : serangan terhadapat kerahasian dan otentikasi,

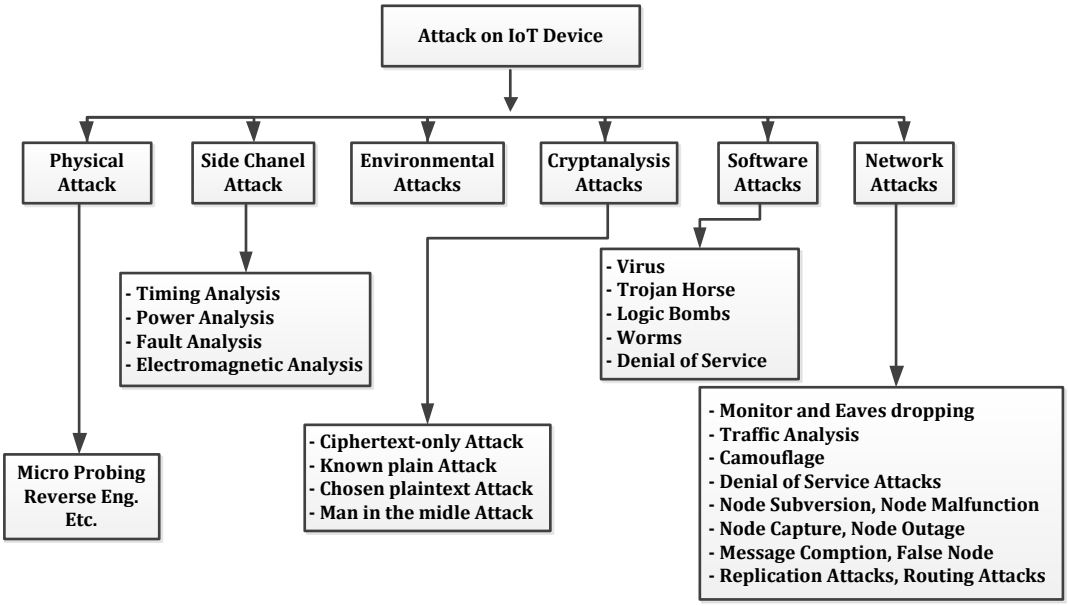
serangan terhadap integritas layanan, dan serangan terhadap ketersediaan jaringan. Jenis serangan yang masuk dalam ketiga kategori ini adalah denial of service (DoS), yaitu serangan yang menyebabkan pengguna yang sah tidak dapat mengakses informasi [7]. Serangan ini dapat terjadi pada berbagai lapisan jaringan WSN [8]:

1. DoS pada lapisan fisik. Lapisan fisik sebagai lapisan yang menjalankan fungsi modulasi, demodulasi, enkripsi, pembangkit frekuensi pembawa, pengirim dan penerima data, dapat diserang dengan cara *jamming* dan *node tampering*. Dengan jamming, penyerang menduduki kanal komunikasi sehingga menghalangi jalur komunikasi antar node. Sementara dengan node tampering, node secara fsik dirusak atau diubah untuk mendapatkan informasi sensitif.
2. DoS pada lapisan link. Lapisan link bertanggung jawab dalam melakukan multipleks berbagai aliran data, mendeteksi bingkai data serta melakukan MAC dan error control. Serangan DoS pada lapisan link dapat dilakukan dengan cara kolisi. Kolisi dipicu dengan mengirimkan paket data secara serempak dari dua node pada kanal frekuensi yang sama [9]. Tubrukan ini akan menyebabkan perubahan kecil pada paket data, sehingga tidak teridentifkasi dengan benar. DoS pada lapisan link juga dapat dilakukan dengan serangan *unfairness* yaitu tubrukan yang dilakukan secara terus menerus. Serangan DoS yang menyebabkan trafik yang tinggi pada kanal ini, menyebabkan aksesibilitas terhadap kanal menjadi sangat terbatas dan menghabiskan baterai sensor.
3. DoS pada lapisan jaringan. Beberapa jenis serangan yang menyebabkan DoS, dapat dilakukan pada lapisan yang bertanggung jawab terhadap proses routing ini [7]. Yang petama adalah spoofng, replaying dan trafk yang salah arah. Serangan kedua adalah Hello flood : serangan ini dilakukan dengan cara membanjiri kanal dengan sejumlah besar pesan yang tidak bermanfaat. Cara ketiga adalah homing. Penyerang mencari di dalam trafk, kepala cluster dan key manager yang memiliki kemampuan untuk mematikan seluruh jaringan. Serangan ke empat adalah serangan Sybil, penyerang mereplikasi sebuah node dan mengenalkannya pada node-node lain dengan identitas yang berbeda-beda. Serangan ke lima adalah *wormhole* yang menyebabkan berpindahnya bit-bit data dari posisi sebenarnya di dalam jaringan. Cara terakhir adalah dengan acknowledgement flooding, dimana node penyerang mencurangi *acknowledgment* dengan memberikan informasi yang salah pada node tujuan.
4. Serangan DoS pada lapisan transport. Lapisan transport berfungsi untuk menjaga kehandalan transmisi datadan mencegah kemacetan akibat tingginya trafk dirouter. Penyerang dapat melakukan serangan flooding yaitu dengan sengaja membanjiri kanal komunikasi dengan trafik yang tinggi, dan serangan desinkronisasi yaitu mengirimkan permintaan pada endpoint untuk mengirim ulang pesan yang salah yang sebetulnya tidak ada. Desinkronisasi akan menyebabkan node kehabisan energi.

5. Serangan DoS pada lapisan aplikasi. Lapisan ini bertanggung jawab terhadap manajemen trafik dan penyedia perangkat lunak bagi aplikasi yang berbeda dengan menterjemahkan data menjadi bentuk yang dapat dipahami [10]. Serangan DoS di lapisan ini diinisiasi dengan menstimulasi sensor node untuk membuat trafik yang sangat besar pada rute menuju base station [9].

2.4.1 Serangan pada perangkat *Internet of Things* (IoT)

Domain serangan keamanan pada perangkat *Internet of Things* (IoT) meningkat dari hari ke hari. Berikut gambar 2.8 merangkum serangan terhadap sistem IoT. [Ramjee Prasad,.dkk. 2011].



Gambar 2.7 Serangan terhadap perangkat IoT

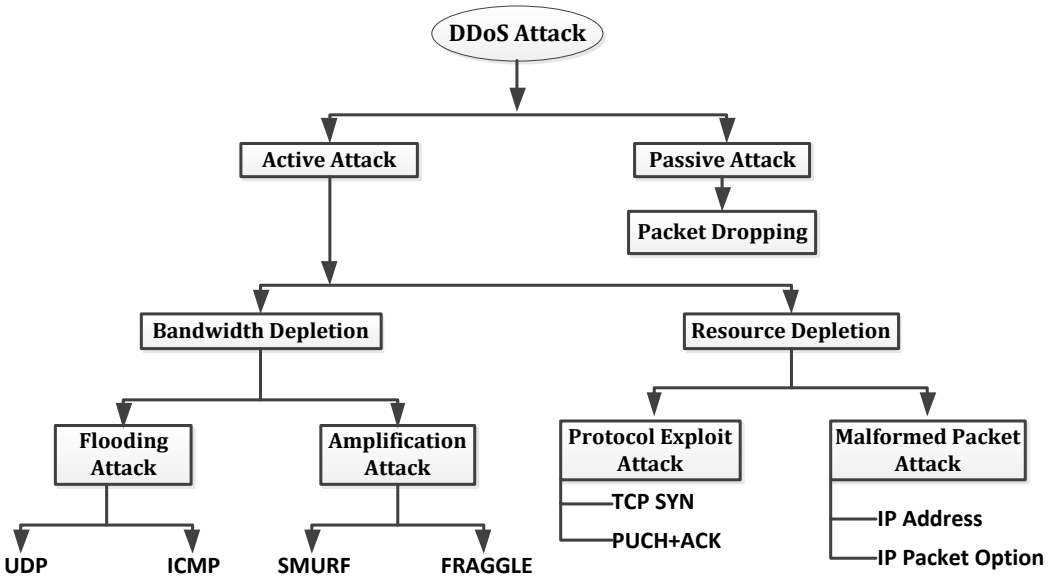
Serangan cyber pada perangkat IoT telah diklasifikasikan ke dalam beberapa kelas seperti yang telah dibahas dalam penelitian (Atamli,.dkk.2014), (Huuck R,.dkk.2015) sebagai berikut :

1) Merusak Jaringan

Musuh dapat memodifikasi perangkat dan memasukkan penipu ke sistem. Oleh karena itu, perangkat tidak akan berfungsi sebagaimana seharusnya. Serangan semacam ini biasanya digunakan untuk mencuri informasi dan menyalahgunakan perangkat lunak dan perangkat keras perangkat IoT.

2) Denial of Service (DoS)

DoS dapat dilakukan dengan menyalahgunakan perangkat, memanipulasi perangkat lunak dan aplikasi, atau mengganggu saluran komunikasi (Atamli,.dkk.,2014). Salah satu serangan DoS adalah serangan gangguan di mana saluran komunikasi mampu menonaktifkan saluran komunikasi sensor dari membawa sinyal dengan menghasilkan tabrakan. Tabrakan akan menyebabkan pesan komunikasi terganggu.



Gambar 2.8 *DDoS Attacks*

3) DoS Terdistribusi

Lihatlah serangan Mirai. Malware Mirai dirancang untuk mengeksploitasi kerentanan yang ada dalam perangkat IoT untuk serangan DDoS. Ada jutaan perangkat IoT di pasar yang salah dikonfigurasi dan diatur untuk meneruskan pesan melalui Transmission Control Protocol (TCP) (Walsh Ray, 2016).

4) Spoofing

Menurut Felten et al *spoofing* dapat didefinisikan sebagai "Teknik yang digunakan untuk memperoleh akses yang tidak sah ke suatu komputer atau informasi, dimana penyerang berhubungan dengan pengguna dengan berpura-pura memalsukan bahwa mereka adalah *host* yang dapat dipercaya". Dalam bahasa *networking*, *spoofing* berarti pura-pura berlaku atau menjadi sesuatu yang sebenarnya bukan.

Musuh menggunakan informasi kredensial yang dimiliki oleh orang lain untuk mendapatkan akses ke layanan yang tidak dapat didekati. Kredensial ini dapat ditemukan dari perangkat itu sendiri, menguping di saluran komunikasi, atau dari kegiatan pengintaian.

5) Pelanggaran Privasi

Musuh dapat mengumpulkan informasi data pribadi dari berbagai sumber yang berbeda, misalnya, meta-informasi dan investigasi aktivitas target serangan.

6) Buffer Overflow

Menggunakan serangan semacam ini, *buffer overflow* memungkinkan musuh untuk mengendalikan atau mematikan prosesor untuk mengubah variabel intinya. Jika program cukup istimewa, maka musuh dapat mengendalikan tuan rumah sehingga dapat dikontrol dan dikendalikan.

7) SQL Injection

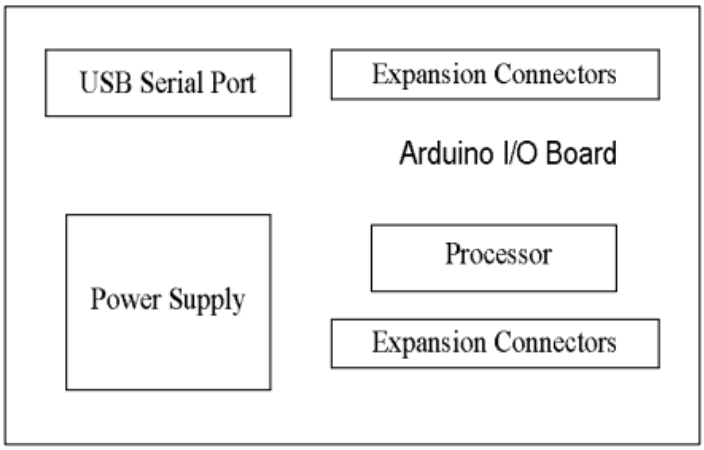
Metode injeksi kode berbahaya yang digunakan untuk menyerang aplikasi yang digerakkan oleh informasi, memanipulasi kelemahan keamanan dalam perangkat lunak aplikasi, mengizinkan musuh untuk spoof identitas, memodifikasi data yang dapat menyebabkan masalah penolakan.

Studi kasus serangan lainnya adalah pada sistem pemantauan glukosa untuk pasien diabetes. Seperti yang dilaporkan pada bulan Oktober 2016, anak perusahaan Johnson & Johnson, Animas, memproduksi perangkat membaca kadar glukosa darah pengguna melalui meter sebelum pompa menggunakan pembacaan ini dengan "berkomunikasi secara nirkabel" dalam pita 900 MHz untuk mengirimkan insulin. Salah satu kelemahan keamanan utama ada kurangnya enkripsi antara komponen-komponen ini. Ini membuka pintu bagi penyadap untuk menangkap informasi seperti data dosis dan hasil glukosa darah. Penyerang dapat dengan mudah mengendus remote / pompa dan kemudian spoof menjadi remote atau pompa. Kerentanan lain adalah saluran komunikasi di mana itu terjadi antara pompa dan meter tidak memiliki timestamps atau nomor urut dan karena ini, tidak ada pertahanan terhadap serangan replay (Charlie Osborne, 2016).

2.5 Arduino Uno

Arduino adalah platform pembuatan prototype elektronik yang bersifatopen-source hardware yang berdasarkan pada perangkat keras dan perangkat lunak yang fleksibel dan mudah digunakan, pengendali mikro single-board, diturunkan dari wiring platform, dirancang untuk memudahkan penggunaan elektronik dalam berbagai bidang. Hardware dalam arduino memiliki prosesor Atmel AVR dan menggunakan software dan bahasa sendiri.

Platform arduino terdiri dari arduino board, shield, bahasa pemrograman arduino, dan arduino development environment. Arduino board biasanya memiliki sebuah chip dasar mikrokontroler Atmel AVR ATmega8 berikut turunannya. Blok diagram arduino board yang sudah disederhanakan dapat dilihat pada gambar 2.8. Shield adalah sebuah papan yang dapat dipasang diatas arduino board untuk menambah kemampuan dari arduino board.



Gambar 2.9 Blok Diagram *Arduino Board*

2.5.1 Sejarah arduino

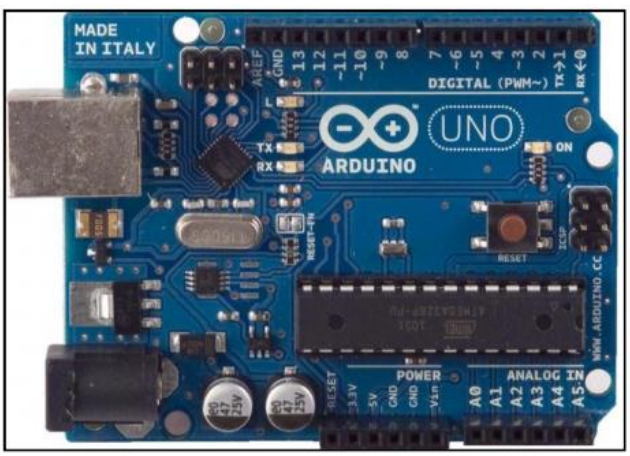
Proyek Arduino dimulai pertama kali di Ovre, Italy pada tahun 2005. Tujuan proyek ini awalnya untuk membuat peralatan control interaktif dan modul pembelajaran bagi siswa yang lebih murah dibandingkan dengan prototype yang lain. Pada tahun 2010 telah terjual dari 120 unit Arduino. Arduino yang berbasis open source melibatkan tim pengembang. Pendiri arduino itu Massimo Banzi dan David Cuartielles, awalnya mereka memberi nama proyek itu dengan sebutan arduino dari ivrea tetapi seturut perkembangan zaman nama proyek itu diubah menjadi Arduino.

Arduino dikembangkan dari thesis hernando Barragan di desain interaksi institute Ivrea. Arduino dapat menerima masukan dari berbagai macam sensor danjuga dapat mengontrol lampu, motor dan aktuator lainnya. Mikrokontroler pada board arduino di program dengan menggunkan bahasa pemrograman arduino (based on wiring) dan IDE arduino (based on processing). Proyek arduino dapatberjalan sendiri atau juga bisa berkomunikasi dengan software yang berjalan pada komputer.

2.5.2 Hardware

Arduino Uno adalah board berbasis mikrokontroler pada ATmega328 yang memiliki 14 pin digital input / output (dimana 6 pin dapat digunakan sebagai output PWM), 6 input analog, 16 MHz osilator kristal, sebuah koneksi USB,sebuah konektor sumber tegangan, sebuah header ICSP, dan sebuah tombol reset.

Hardware dalam arduino memiliki beberapa jenis, yang mempunyai kelebihan dan kekurangan dalam setiap papannya. Penggunaan jenis arduinodisesuaikan dengan kebutuhan, hal ini yang akan mempengaruhi dari jenis prosesor yang digunakan. Jika semakin kompleks perancangan dan program yang dibuat, maka harus sesuai pula jenis kontroler yang digunakan. Yang membedakan antara arduino yang satu dengan yang lainnya adalah penambahan fungsi dalam setiap papan circuitnya dan jenis mikrokontroler yang digunakan. Dalam penelitian ini, jenis arduino yang digunakan adalah arduino uno, dengan gambar sebagai berikut:



Gambar 2.10 *Hardware Arduino Uno*

Hardware arduino uno memilki spesifikasi sebagai berikut :

a. 14 pin IO Digital (pin 0–13). Sejumlah pin digital dengan nomor 0–13 yang dapat dijadikan input atau output yang diatur dengan cara membuat program IDE.
b. 6 pin Input Analog (pin 0–5).Sejumlah pin analog bernomor 0–5 yang dapat digunakan untuk membaca nilai input yang memiliki nilai analog dan mengubahnya ke dalam angka antara 0 dan 1023.
c. 6 pin Output Analog (pin 3, 5, 6, 9, 10 dan 11). Sejumlah pin yang sebenarnya merupakan pin digital tetapi sejumlah pin tersebut dapat diprogram kembali menjadi pin output analog dengan cara membuat programnya pada IDE.

Papan Arduino Uno dapat mengambil daya dari USB port pada computer dengan menggunakan USB charger atau dapat pula mengambil daya dengan menggunakan suatu AC adapter dengan tegangan 9 volt. Jika tidak terdapat power supply yang melalui AC adapter, maka papan Arduino akan mengambil daya dari USB port. Tetapi apabila diberikan daya melalui AC adapter secara bersamaan dengan USB port maka papan Arduino akan mengambil daya melalui AC adapter secara otomatis.

2.5.3 Software arduino

Software arduino yang digunakan adalah driver dan IDE (*Integrated Development Environment),* walaupun masih ada beberapa software lain yang sangat berguna selama pengembangan arduino. IDE (*Integrated Development Environment)* suatu program khusus untuk suatu komputer agar dapat membuat suatu rancangan atau sketsa program untuk papan Arduino. IDE (*Integrated Development Environment)* arduino merupakan software yang sangat canggih ditulis dengan menggunakan java. Secara bahasa mudahnya merupakan lingkungan terintegrasi yang digunakan untuk melakukan pengembangan. Disebut sebagai lingkungan karena melalui software inilah Arduino dilakukan pemrograman untuk melakukan fungsi-fungsi yang dibenamkan melalui sintaks pemrograman

Arduino menggunakan bahasa pemrograman sendiri yang menyerupai bahasa C. IDE (*Integrated Development Environment)* arduino terdiri dari:

1. ***Editor Program***. Sebuah window yang memungkinkan pengguna menulis dan mengedit program dalam bahasa processing
2. ***Compiler***. Sebuah modul yang mengubah kode program menjadi kode biner bagaimanapun sebuah mikrokontroler tidak akan bisa memahami bahasa processing.
3. ***Uploader***. Sebuah modul yang memuat kode biner dari komputer ke dalam memory di dalam papan arduino.

Dalam bahasa pemrograman arduino ada tiga bagian utama yaitu struktur, variabel dan fungsi (Artanto, 2012) :

1. Struktur Program Arduino

a. Kerangka Program

Kerangka program arduino sangat sederhana, yaitu terdiri atas dua blok.Blok pertama adalah void setup () dan blok kedua adalah void loop.

1) Blok Void setup ()

Berisi kode program yang hanya dijalankan sekali sesaat setelah arduino dihidupkan atau di-reset. Merupakan bagian persiapan atau instalasi program.

2) Blok void loop()

Berisi kode program yang akan dijalankan terus menerus. Merupakan tempat untuk program utama.

b. Sintaks Program

Baik blok void setup loop () maupun blok function harus diberi tanda kurung kurawal buka "{"sebagai tanda awal program di blok itu dan kurung kurawal tutup"}" sebagai tanda akhir program.

2. Variabel

Sebuah program secara garis besar dapat didefinisikan sebagai instruksi untuk memindahkan angka dengan cara yang cerdas dengan menggunakan sebuah varibel.

3. Fungsi

Pada bagian ini meliputi fungsi input output digital, input output analog, advanced I/O, fungsi waktu, fungsi matematika serta fungsi komunikasi. Pada proses Uploader dimana pada proses ini mengubah bahasa pemrograman yang nantinya dicompile oleh avr-gcc (avr-gcc compiler) yang hasilnya akan disimpan kedalam papan arduino.

Avr-gcc compiler merupakan suatu bagian penting untuk software bersifat open source. Dengan adanya avr-gcc compiler, maka akan membuat bahasa pemrogaman dapat dimengerti oleh mikrokontroler. Proses terakhir ini sangat penting, karena dengan adanya proses ini maka akan membuat proses pemrogaman mikrokontroler menjadi sangat mudah.

Berikut ini merupakan gambaran siklus yang terjadi dalam melakukan sebuah pemrogaman berbasis Arduino :

1. Koneksikan papan Arduino dengan komputer melalui USB port.
2. Tuliskan sketsa rancangan suatu program yang akan dimasukkan ke dalam papan Arduino.
3. Upload sketsa program ke dalam papan Arduino melalui kabel USB dankemudian tunggu beberapa saat untuk melakukan restart pada papan Arduino. Papan Arduino akan mengeksekusi rancangan sketsa program yang telah dibuat dan di-upload ke papan Arduino.

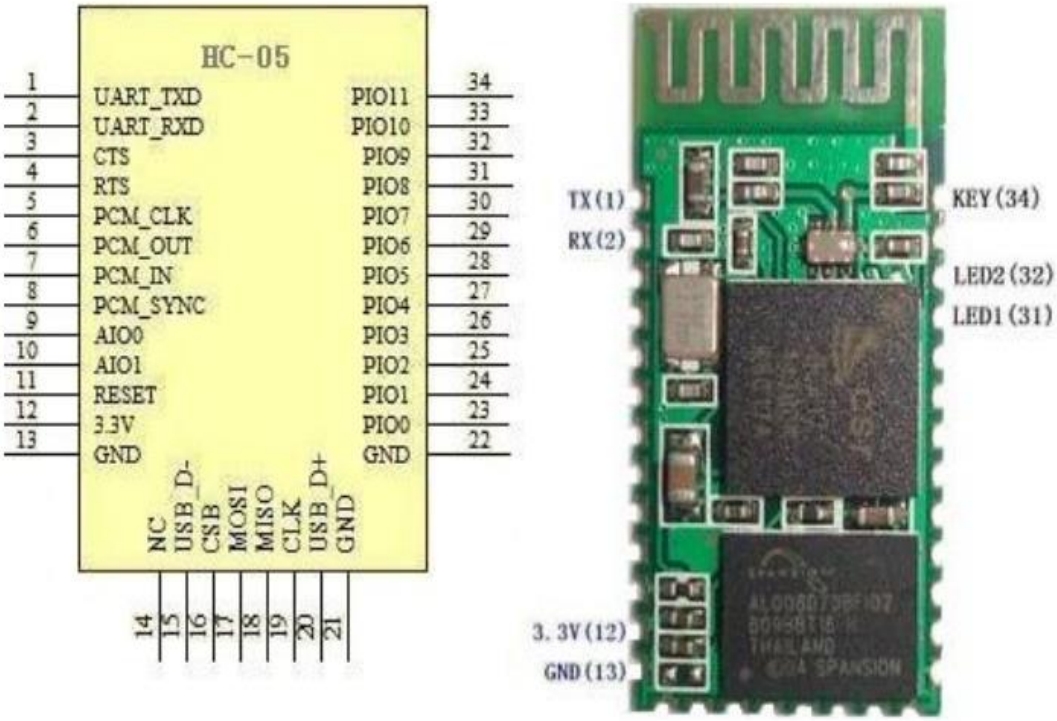
2.6 Bluetooth HC-05

Bluetooth HC-05 adalah sebuah modul Bluetooth SPP (Serial Port Protocol) yang mudah digunakan untuk komunikasi serial wireless (nirkabel) yang mengkonversi port serial ke Bluetooth. HC-05 menggunakan modulasi bluetooth V2.0 + EDR (Enchanced Data Rate) 3 Mbps dengan memanfaatkan gelombang radio berfrekuensi 2,4 GHz. Modul ini dapat digunakan sebagai slave maupun master. HC-05 memiliki 2 mode konfigurasi, yaitu AT mode dan Communication mode. AT mode berfungsi untuk melakukan pengaturan konfigurasi dari HC-05. Sedangkan Communication mode berfungsi untuk melakukan komunikasi bluetooth dengan piranti lain.

Dalam penggunaannya, HC-05 dapat beroperasi tanpa menggunakan driver khusus. Untuk berkomunikasi antar Bluetooth, minimal harus memenuhi dua kondisi berikut :

1. Komunikasi harus antara master dan slave.
2. Password harus benar (saat melakukan pairing).

Jarak sinyal dari HC-05 adalah 30 meter, dengan kondisi tanpa halangan.



Gambar 2.11 Bentuk fisik blutooth HC-05 dan pin out

Bluetooth HC-05 memiliki spesifikasi dalam penggunaannya antara lain:

- Sensitivitas -80dBm (Typical)
- Daya transmit RF sampai dengan +4dBm.
- Operasi daya rendah 1,8V - 3,6V I/O.
- Kontrol PIO.
- Antarmuka UART dengan baudrate yang dapat diprogram.

Bluetooth HC-05 memiliki command set dalam melakukan perubahan baud rate, nama Bluetooth, perubahan password dan yang lainnya dengan memanfaatkan jalur TX dan RX. Konfigurasi dilakukan pada pc dengan menggunakan hyper terminal dan bluetooth yang sudah terkoneksi dengan PC (personal computer) yang telah melalui rs232. Berikut adalah command set utama yang digunakan antara lain :

1. Command "AT"

Command AT digunakan untuk melakukan test bluetooth. Untuk mengetahui jika bluetooth dapat berfungsi atau tidak, ketika command "AT" dikirimkan maka akan mendaatkan respon balik, atau bluetooth akan mengirimkan command "OK" melalui jalur TX Bluetooth.

2. Command "AT+BAUD"

Untuk melakukan perubahan baud rate yang digunakan dengan mengirimkan "AT+BAUD". Sebagai contoh "AT+BAUD1", "1" setelah baud mengartikan baud rate yang digunakan. Baud rate yang disediakan oleh Bluetooth yaitu:

a. (1200) d. (9600) g. (57600)
b. (2400) e. (19200) h. (115200)
c. (4800) f. (38400)

Respon yang akan diterima ketika proses penggantian baud rate selesai yaituBluetooth akan mengirimkan "OK" melalui jalur TX Bluetooth.

3. Command "AT+NAME (defice name)"

Command "AT+NAME (defice name)" digunakan untuk melakukan perubahan nama device bluetooh, sebagai contoh "AT+NAMETEST" yang berarti bahwa Bluetooth tersebut bernama test ketika di deteksi oleh perangkat lain. Ketika command telah berhasil dikirmkan maka respon balik yang kan di dapatkan adalah "OK set name" namun, jika tidak berhasil atau gagal maka respon yang diterima adalah "FAIL"

4. Command "AT+PINxxxx"

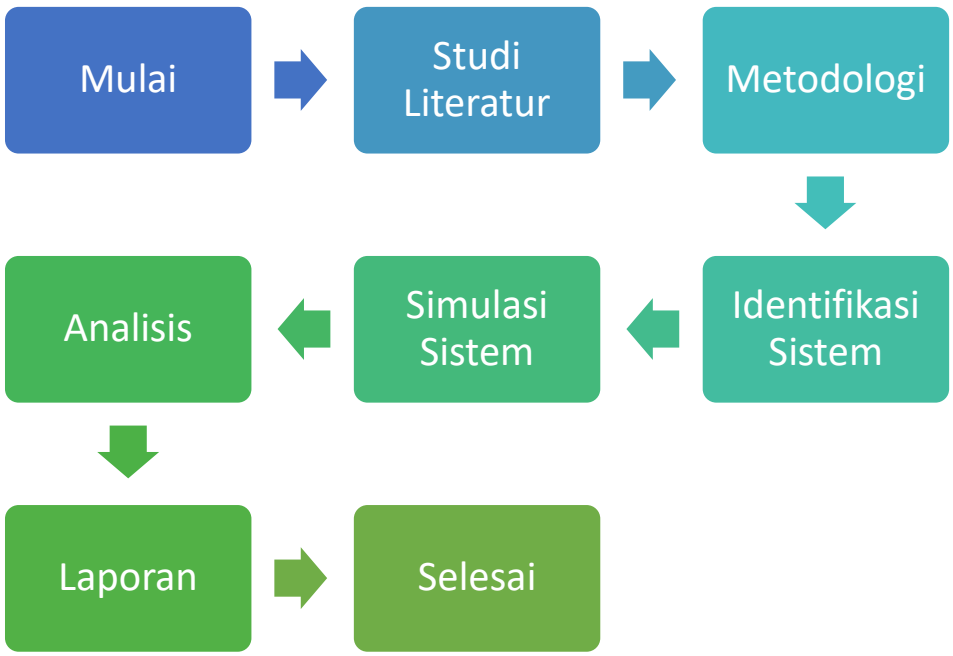
Command "AT+PINxxxx" digunakan untuk melakukan perubahan pin. Pin Bluetooth akan muncul ketika hardware lain akan melakukan koneksi ke Bluetooth HC-05. Proses setting hanya bisa dilakukan pada saat Bluetooh module dalam kondisi tidak terhubung/paired dengan device lain, hal ini bias dilihat dari nyala led pada modul. Jika led menyala berkedip berarti Bluetooth module ini tidak terkoneksi dengan device bluetooth lain.

5. Command "AT+VERSION"

Command "AT+VERSION" digunakan untuk mengetahui versi Bluetooth. Ketika command dikirimkan maka bluetooh akan mengirim respon balik yaitu dengan mengirmkan versi bluetoothnya, jika blueooh yang digunakan adalah HC06 maka responn yang dikirmkan adalah "Linvor1.5".

Bab 3 Metodologi Penelitian

Bab ini menjelaskan bagaimana cara penelitian ini dilakukan, sehingga dapat memberikan rincian tentang alur atau langkah-langkah yang dibuat secara sistematis serta dapat digunakan dengan jelas dalam menyelesaikan masalah, membuat analisa terhadap hasil penelitian. Adapun tahapan penelitian ini dapat dilihat pada gambar 3.1.



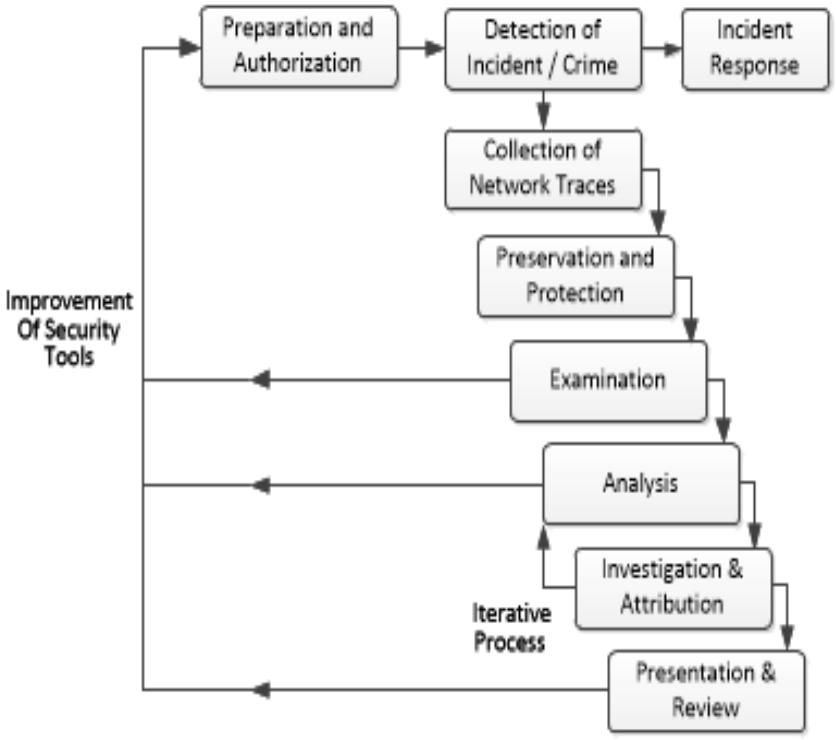
Gambar 3.1 Alur Metodologi Penelitian

3.1 Literatur Review

Literatur review dilakukan untuk mendapatkan informasi sumber mengenai topik-topik yang akan diteliti yang dapat diperoleh dari buku, dokumen, artikel, atau bahan tertulis lainnya yang berupa buku laporan, teori, maupun penemuan lainya yang bersifat *online* maupun *offline* yang bertujuan memberikan sumber informasi.

Sedangkan *review* atau kajian pustaka dilakukan untuk tujuan terhadap dilkukannya penelitian yang terkait dengan masalah-masalah yang terkait dengan *network forensics* untuk mendeteksi serangan *flooding* dengan menggunakan metode *Generic Framework for Network Forensics* pada perangkat *Internet of Things* (IoT).

Berikut juga metode forensik jaringan yang digunakan agar dapat menunjang tujuan akhir dalam penelitian ini.



Gambar 3.2 Metode *Generic Framework for Network Forensics*

3.2 Identifikasi Sistem

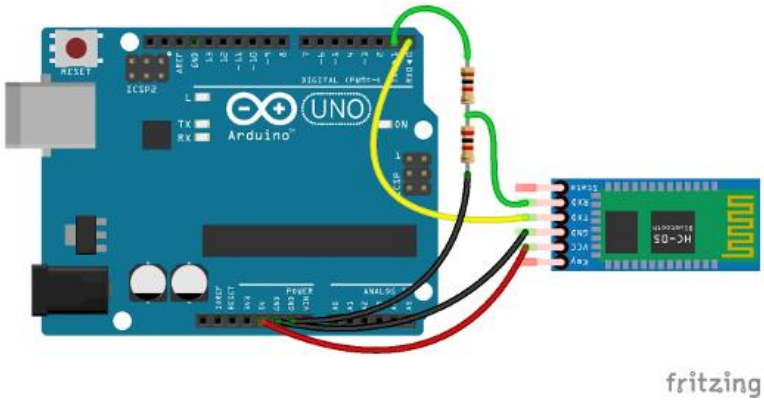
Merupakan tahap perancangan dan implementasi sistem forensik jaringan untuk mendeteksi serangan *flooding* dengan metode *Generic Framework for Network Forensics* pada perangkat *Internet of Things* (IoT) yang akan digunakan sebagai objek penelitian. Terdiri dari beberapa komponen berupa :

Gambar 3.3 Identifikasi sistem perangkat IoT

3.3 Konfigurasi Ardiuno UNO dan Bluetooth Hc-05

Mempersiapkan paket arduino yang merupakan paket utama yang dibutuhkan dalam sistem, paket yang digunakan adalah paket driver arduino yang dapat diinstal langsung. Melakukan konfigurasi *arduino* dengan *Bluetooth Hc-05* merupakan konfigurasi awal dengan tujuan untuk mendeteksi dan menganalisis data *Traffic log file* yang terdapat dalam perangkat dan memastikan bluetooth telah berfungsi dengan baik.

Berikut adalah sekema konfigurasi Arduino UNO dengan Bluetooth HC-05 yang akan digunakan dalam penelitian forensik perangkat *Internet of Things* (IoT) :



Gambar 3.4 Skema Konfigurasi Arduino dengan Blutooth HC-05

Untuk melakukan konfigurasi bluetooth HC05, hal pertama yang harus kita lakukan adalah menghubungkan modul bluetooth ke PC atau laptop dengan bantuan modul USB. Beberapa konfigurasi dalam Arduino yaitu menguhubungkan Arduino dengan komputer, melakukan komunikasi serial seperti mengirimkan dan menerima data sensor melalui serial terminal pada Arduino IDE melalui USB Connector**.** Power Jack **:** Tegangan input untuk menghidupkan Arduino, IC ATMEGA328p **:** IC Microcontroler keluaran ATMEL dengan booth loader Arduino. I/O Digital dipergunakan untuk input dan output digital, pada pin 3,5,6,9,10,11 memiliki tanda (~) menunjukan bahwa pin tersebut selain memiliki fasilitas I/O Digital juga memiliki PWM (Pulse Width Modulation) dengan rentang nilai output sebesar 8 bit atau setara dengan nilai antara 0-255. Selanjutnya adalah Input Analog yang digunakan untuk input data sensor, potensiometer dan perangkat analog input lainya. Kemudian Power **y**ang digunakan untuk mengambil power 5V, 3.3V, GND.

Konfigurasi juga dilakukan pada device bluetooth HC-05. Pada saat melakukan konfigurasi bluetooth maka posisi bluetooth dalam keadaan tidak berhubungan dengan perangkat arduino yang menggunakan nirkabel. Sehingga akan benar-benar dipastikan bahwa bluetooth dalam keadaan aktif tanpa koneksi. Selanjutnya dilakukan Pengaturan Default bluetooth yaitu Baudrate : 9600 bps Nama : linvor Kode Pairing : 1234. Setiap perubahan konfigurasi di atas akan disimpan bahkan ketika power dimatikan. Semua perintah yang dikirim ke Buetooth tidak harus dengan karakter baris baru. Oleh karena itu kami sarankan untuk menggunakan 'Serial Monitor pada Arduino IDE untuk mengkonfigurasi modul Bluetooth. Selanjutnya prosedur untuk melakukannya konfigurasi bluetooth yaitu menghubungkan Bluetooth ke PC, LED harus berkedip, membuka software Arduino IDE, memilih port COM yang benar bahwa Bluetooth terhubung.

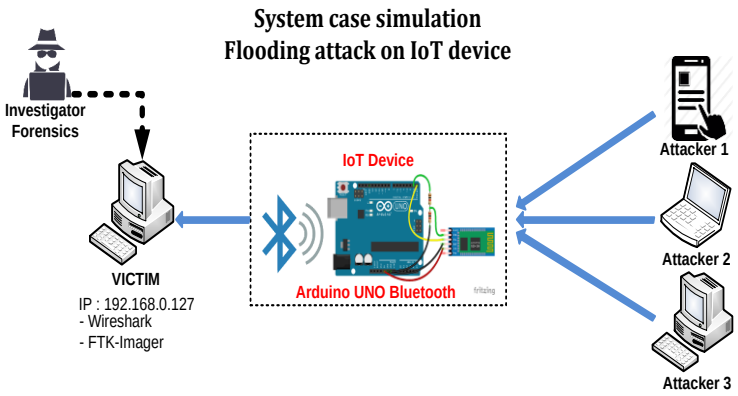
3.4 Simulasi Kasus

Merupakan tahap dilakukannya simulasi sistem untuk mencoba mengimplementasikan forensika jaringan pada perangkat *Internet of Things* (IoT). Simulasi sistem bertujuan untuk melakukan pengujian forensika jaringan dengan metode *network forensics process model* terhadap perangkat *Internet of Things* (IoT) Arduino dalam mendeteksi serangan flooding.

Dalam sebuah serangan *Denial of Service*, si penyerang akan mencoba untuk mencegah akses seorang pengguna terhadap sistem atau jaringan dengan menggunakan beberapa cara, yakni sebagai berikut :

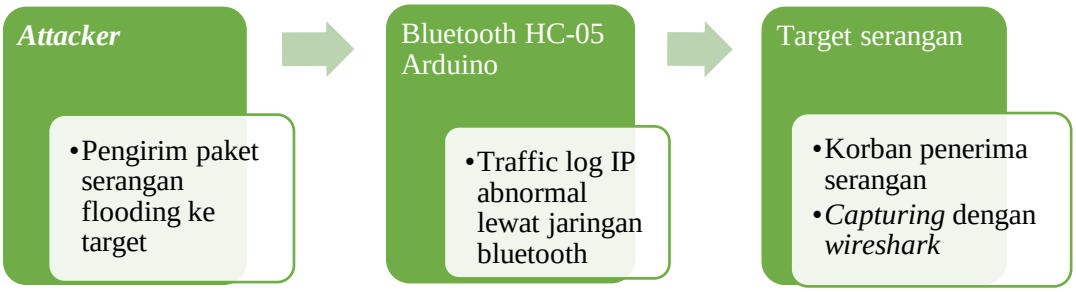
- Membanjiri lalu lintas jaringan dengan banyak data sehingga lalu lintas jaringan yang datang dari pengguna yang terdaftar menjadi tidak dapat masuk ke dalam sistem jaringan. Teknik ini disebut sebagai *traffic flooding*.
- Membanjiri jaringan dengan banyak request terhadap sebuah layanan jaringan yang disedakan oleh sebuah host sehingga request yang datang dari pengguna terdaftar tidak dapat dilayani oleh layanan tersebut. Teknik ini disebut sebagai *request flooding*.
- Mengganggu komunikasi antara sebuah host dan kliennya yang terdaftar dengan menggunakan banyak cara, termasuk dengan mengubah informasi konfigurasi sistem atau bahkan perusakan fisik terhadap komponen dan server.

Berikut adalah gambar simulasi sistem kasus serangan flooding terhadap perangkat *Internet of Things* (IoT) :



Gambar 3.5 Simulasi sistem serangan flooding pada Perangkat IoT

Skema serangan adalah ketika *attacker* mengirimkan paket *flood* kepada target maka pengiriman paket tersebut akan menyebabkan aktifitas target jadi lambat, bahkan server akan mati ketika pengiriman serangan paket tersebut melebihi beban yang dimiliki. Gambar 3.4 menunjuukan tahapan simulasi kasus deteksi serangan flooding pada perangkat IoT.



Gambar 3.6 Tahap simulasi serangan flooding pada perangkat IoT

3.5 Analisis Investigasi Forensik

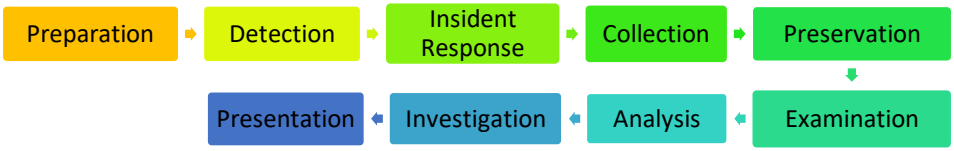
Menurut Muh Al Azhar forensik adalah suatu proses ilmiah atau suatu usaha ilmiah yang didasari ilmu pengetahuan dalam memngumpulkan, menganalisa dan menghadirkan bukti dalam suatupersidangan di pengadilan untuk membantu pengungkapan suatu kejahatan melalui pengungkapa bukti-bukti yang sah menurut undang-undang dan peraturan yang berlaku. Digital forensik merupakan aplikasi bidang ilmu pengetahuan dan teknologi komputer untuk kepentingan pembuktian hukum (pro justice), yang dalam hal ini adalah untuk membuktikan kejahatan berteknologi tinggi atau computer crime secara ilmiah (scientific) sehingga bisa mendapatkan bukti-bukti digital yang dapat digunakan untuk menjerat pelaku kejahatan tersebut. Disinilah tugas untuk para investigator dalam menangani kasus penyelidikan untuk dapat merecover ulang kejadian peristiwa tindak kriminal.

Tahap dilakukannya tahapan analisa model proses forensik jaringan dengan metode *Generic Framework for Network Forensics* terhadap perangkat *Internet of Things* (IoT). Investigasi ini dilakukan dengan 9 tahap penelitian berupa :

1) Tahap *Preparation*, yaitu langkah awal persiapan dalam melakukan investigasi. Persiapan ini meliputi alat yang berupa hardware dan software, serta sumber daya manusia yang akan melakukan kegiatan investigasi.
2) Tahap kedua *Detection* yaitu memastikan adanya pelanggaran hukum.
3) Tahap ketiga yaitu *Incident Respone*, melakukan tindakan awal dalam proses investigasi setelah terkonfirmasi adanya pelanggaran hukum.
4) Tahap ke empat *Collection* yaitu pengumpulan data hasil penelusuran jejak pada jaringan yang terkena serangan.
5) Tahap ke lima adalah *Preservation*, menjaga integritas pada data dan barang bukti elektronik yang telah dikumpulkan, melakukan *chain of custody* dan teknik *hashing*.
6) Tahap ke enam *Examination* adalah pencarian atau pengolahan barang bukti digital dengan mencocokan barang bukti dengan pelanggaran hukum yang terjadi.

7) Tahap ke tujuh *Analysis* adalah tahap analisa pada data yang sudah diolah dengan metode menggali informasi yang ada dalam barang bukti.
8) Tahap delapan *Investigation*, melakukan tahap penelusuran kembali dengan menggunakan informasi yang ada dalam barang bukti digital.
9) Tahap ke sembilan *Presentation* adalah menggambarkan hasil observasi dengan bahasa yang mudah dimengerti dan umum di persidangan.

Model proses forensic jaringan ini dapat dilihat pada gamabr 3.5.



Gambar 3.7 Metode *Generic Framework for Network Forensics*

Dari model proses forensik tersebut untuk memepermudah klarifikasi jenis serangan dan asal serangan dari mana saja akan dikelompokkan menggunakan data analisa sebagai berikut :

No.	Jenis Serangan	IP Address	Timestamp	Source Address	Destinatian Address	Port	Jumlah

Tabel 3.1 Pengelompokan Data

Tampilan pada table 3.1 menunjukan *traffic log file* yang masuk sebagai jenis serangan berupa beberapa file yang harus di klarifikasi untuk menganalisis data barang bukti yang ditemukan. Dari data tersebut nantinya akan ditemukan tempat dimana alamat penyerang, jenis seragan, waktu yang digunakan, darimana asalnya, dan bagaimana itu terjadi yang berguna untuk membuktikan, mengumpulkan dan menganalisis serta melakukan proses investigasi dari segala aktifitas yang melalui perangkat IoT dengan proses *capturing* jaringan menggunakan aplikasi *wireshark*, serta dapat mengetahui dengan grafik diagram serangan yang paling banyak dilakukan pada sistem. Berikut tabel grafik diagram serangan :

No	Nama Serangan	Jumlah Serangan	Total

Tabel 3.2 Diagram Grafik Serangan

3.6 Laporan

Merupakan tahap pembuatan laporan dan hasil pembuktian identifikasi model jaringan forensik untuk menemukan serangan *flooding* berupa traffic logs file jaringan dari perangkat *Internet of Things* (IoT). Laporan berisi mengenai pendahuluan, litelatur review, metodologi penelitian, hasil dan pembahasan, serta penutup.

Kesimpulan yang diperoleh dari penelitian ini akan dimasukkan ke dalam bagian penutup dari laporan, berikut juga saran untuk penelitian-penelitian selanjutnya. Gambar 3.6 menunjukkan tahapan penyusunan laporan dalam penelitian.

Gambar 3.8 Tahapan penyusunan laporan

Laporan yang disusun pada ahirnya diharapkan dapat memberikan gambaran secara meyeluruh mengenai topik penelitian ini, serta dapat memberikan rekomendasi yang bermanfaat untuk penelitian selanjutnya.

Bab 4 Pembahasan

Bab ini membahas tentang langkah-langkah penelitian, analisis dan hasil yang didapatkan dari penelitian ini. Pembahasan dalam bab ini meliputi tahap studi identifikasi system yang digunakan untuk objek penelitian baik berupa hardware ataupun software, tahap konfigurasi digunakan untuk mengkonfigurasi perangkat *Internet of Things* (IoT) berupa *arduino uno* dengan *bluetooth hc-05*, melakukan simulasi kasus serangan flooding pada perangkat *Internet of Things* (IoT). Tahap analisis digunakan untuk mencari barang bukti dari hasil traffic file log pada perangkat dengan metode *Generic Framework for Network Forensics*.

4.1 Literatur Review

Pada tahapan ini dilakukan kajian literature terhadap penelitian terkait *IoT forensics* dengan studi kasus serangan *flooding* terhadap perangkat *Internet of Things* (IoT) dan penggunaan metode *Generic Framework for Network Forensics* yang akan dijadikan landasan teknis dalam penelitian ini. Penelitian yang dilakukan oleh (Lanke & Jacob, 2014) menjelaskan bahwa teknik *flooding* merupakan serangan yang ditunjukkan untuk mengacaukan atau menghentikan sebuah layanan secara bersama-sama. Aktifitas *flooding* merupakan serangan terhadap sebuah komputer atau server di dalam jaringan internet dengan cara menghabiskan sumber (*resource*) yang dimiliki oleh komputer tersebut sampai komputer tersebut tidak dapat menjalankan fungsinya dengan benar sehingga secara tidak langsung mencegah pengguna lain untuk memperoleh akses layanan dari komputer yang diserang tersebut (Lanke & Jacob, 2014).

Selanjutnya pada penelitian Meet Tilva dan Vandana Rohokale (2016) memberikan gambaran tentang model penggunaan forensik jaringan untuk mendeteksi sebuah paket berbahaya yang terinfeksi dan mengidentifikasi sumber paket yang berbahaya pada lingkungan simulasi. Mensimulasikan serangan Denial of Service (DoS) untuk menganalisa perangkat *Internet of Things* (IoT) yang terinfeksi. Memantau dan komunikasi paket pada *contiki coja simulator* untuk mengidentifikasi forensik jaringan. Peneliti mengusulkan penelitian ini dapat diperluas untuk membuat investigasi forensik jaringan yang lebih aman dengan mekanisme berbasis Log untuk membuat sistem lebih aman dan menjadi mudah untuk menemukan kerusakan tertentu pada sistem.

Pada penelitian ini akan dilakukan investigasi forensik terhadap perangkat *Internet of Things* (IoT) berbasis traffic log dengan model proses forensik jaringan.

4.2 Identifikasi Sistem

Tahap identifikasi sistem jaringan pada perangkat *Internet of Things* (IoT) yang akan digunakan sebagai objek penelitian. Terdiri dari beberapa komponen berupa :

1. Kebutuhan Perangkat Keras

Kebutuhan perangkat keras dalam penelitian ini menggunakan beberapa komponen jaringan berupa :

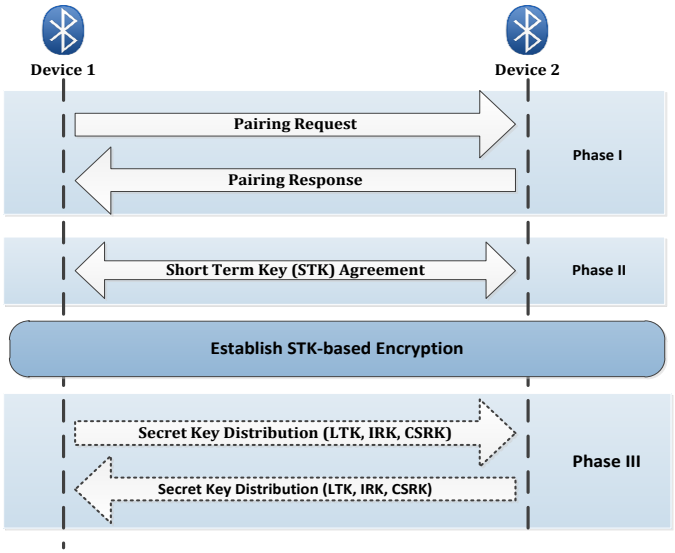
- PC dengan Processor Core i3
- Module Arduino
- USB charger
- Bluetooth HC-05
- Kabel jamper

2. Kebutuhan Perangkat Lunak

Kebutuhan perangkat lunak yang digunakan untuk kebutuhan forensic jaringan dalam penelitian ini adalah :

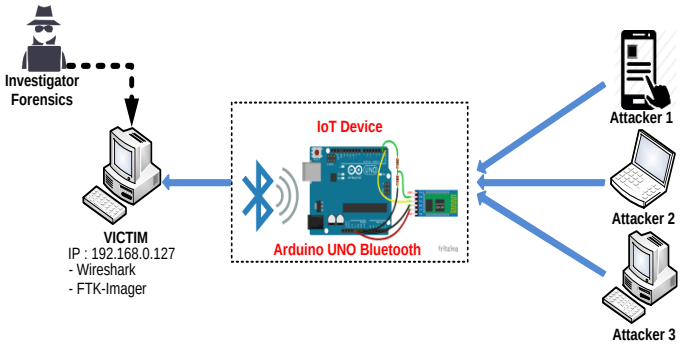
- OS Windows 10
- Driver Arduino dan IDE (*Integrated Development Environment)*
- LOIC 2.0.4-1
- Wireshark 2.4.5

Penelitian forensic jaringan ini menggunakan perangkat IoT Bluetooth Arduino yang bertindak sebagai target serangan pada saat implementasi. *Personal Computer* (PC) target menggunakan IP yang diakses melalui jaringan lokal dengan koneksi jaringan bluetooth. Melakukan pairing PIN code bluetooth supaya antar perangkat terhubung sebelum dilakukan simulasi serangan flooding.



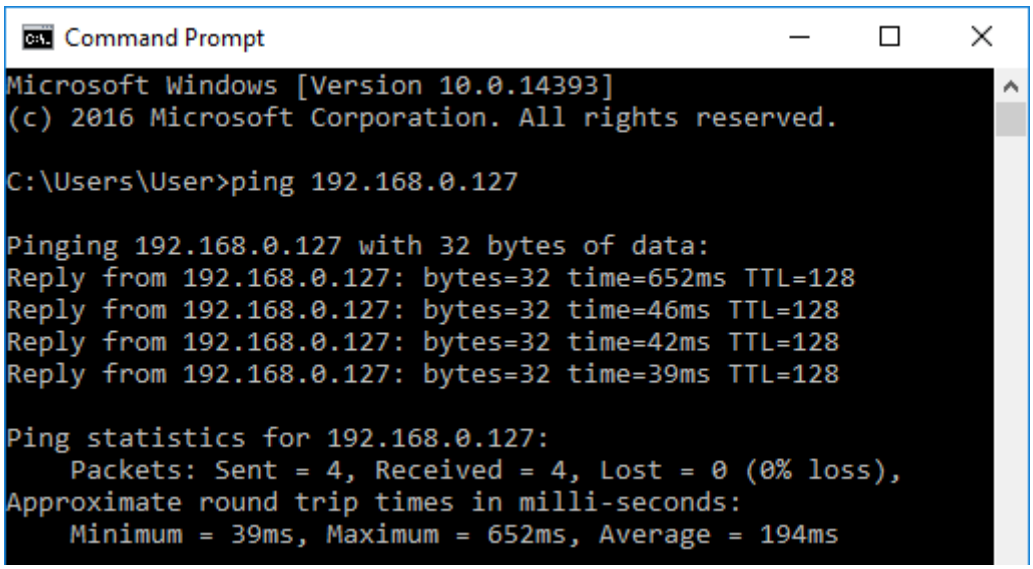
Gambar 4.1 *Pairing* Perangkat Bluetooth

Letak dari investigator forensik jaringan pada penelitian ini dapat dilihat di gambar 4.2 yaitu artsitektur dari forensik jaringan serangan pada perangkat *Internet of Thigs* (IoT).



Gambar 4.2 Arsitektur Forensik Jaringan

Selanjutnya, gambar 4.3 menggambarkan cara percobaan test ping IP dari PC sebagai *attacker* ke PC *victim* menggunakan koneksi jaringan bluetooth dan memastikan bahwa terkoneksi dengan baik.



Gambar 4.3 Test Ping IP victim

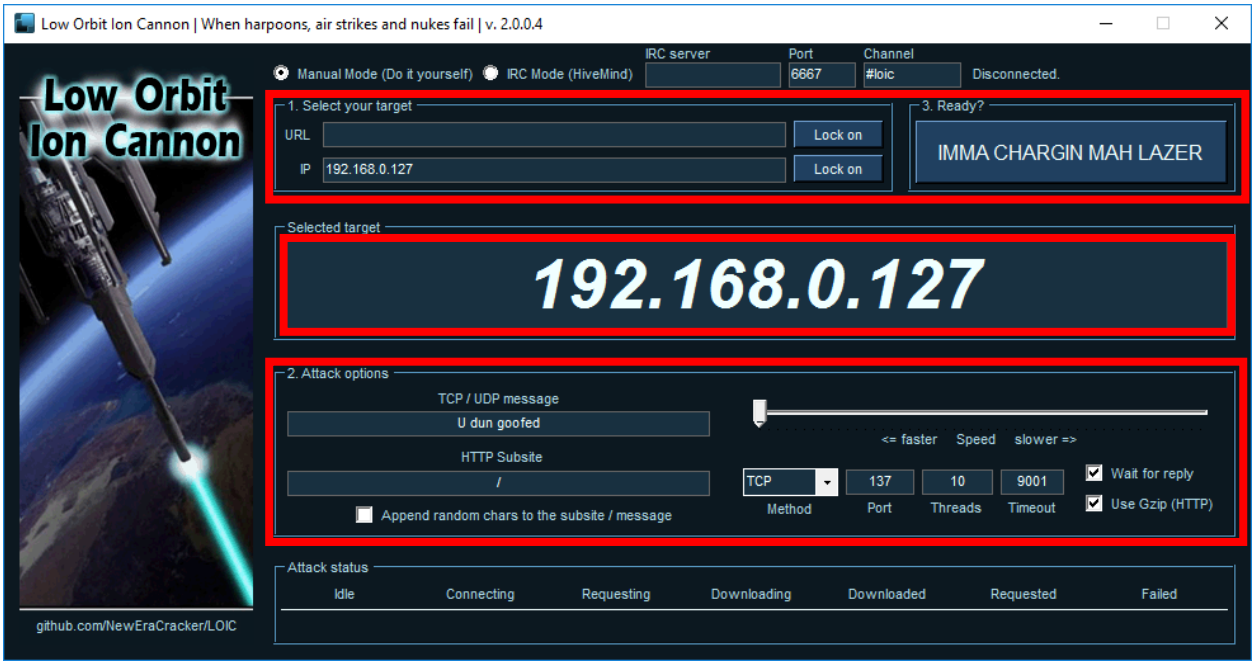
Tahap awal konfigurasi merupakan proses percobaan test Ping IP victim menggunakan cmd dan memastikan terhubung dengan baik sebelum dilakukan simulasi serangan *flooding* menggunakan aplikasi LOIC pada perangkat.

4.3 Simulasi Serangan Flooding

Proses simulasi merupakan tahap awal dalam pengujian untuk mendeteksi serangan *flooding*. Simulasi serangan menggunakan alat bantu LOIC (*Low Orbit Ion Canon*). LOIC (*Low Orbit Ion Canon*) merupakan alat yang digunakan untuk menguji serangan pada perangkat bluetooth IoT.

Alat ini mempunyai kelebihan dapat melakukan pengiriman paket *request* berdasarkan protokol TCP, UDP maupun ICMP. Selain itu target pada *port* yang akan dikirim dapat ditentukan oleh penyerang. Dalam pengujian ini, LOIC digunakan untuk melakukan serangan ke port 137.

Proses pengujian serangan dengan memasukkan alamat IP target pada aplikasi LOIC dari mesin *attacker* pada menu 1 (satu) atau *select your target*, kemudian tetapkan alamat yang akan diserang menggunakan tombol *lock on* yang berada pada menu 1 (satu), selanjutkan menentukan target *port* adalah 137, target protokol adalah UDP, jumlah *threads* yang akan dikirimkan sebanak 10, dan kecepatan pengiriman paket pada tingkat faster pada menu 3 (tiga), Aplikasi LOIC ditunjukkan pada gambar 4.4.



Gambar 4.4 Serangan *Flooding*

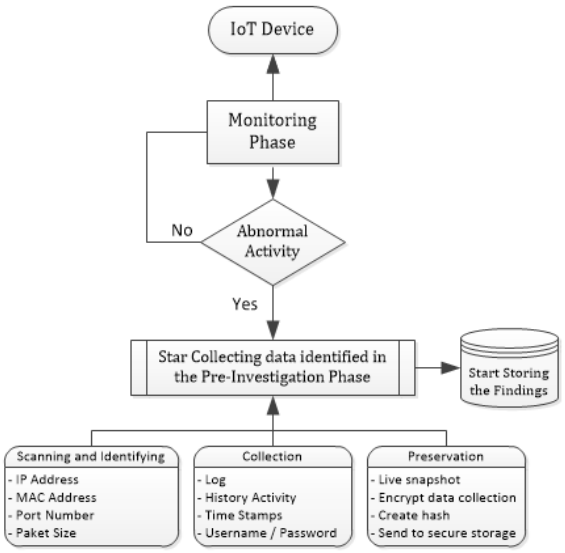
Setelah semua konfigurasi dimasukkan pada aplikasi LOIC kemudian melakukan serangan dengan menekan tombol *start flooding* untuk memulai atau *stop flooding* untuk menghentikan serangan. Serangan *flooding* dilakukan selama 5-10 menit.

4.4 Analisis dan Investigasi Forensik

Pada penelitian investigasi forensik jaringan untuk mendeteksi serangan *flooding* pada perangkat *Internet of Things* (IoT). Menganalisis logs file paket jaringan yang melewati prangkat bluetooth dengan melakukan *capturing* aktifitas jaringan menggunakan aplikasi wireshark. Investigator forensik melakukan analisis pada perangkat *Internet of Things* (IoT) untuk menemukan paket-paket serangan dengan menggunakan model proses forensik jaringan.

4.4.1 Implementasi Forensik Jaringan

Implementasi pada penelitian forensik jaringan terdapat pada rancangan arsitektur forensic jaringan seperti gambar yang ditunjukkan pada gambra 4.5. Yang merupakan arsitektur forensik jaringan pada perangkat IoT dalam mendeteksi serangan *flooding*. Pengambilan data dilakukan oleh peneliti dengan cara melakukan proses *capturing* jaringan dengan alat bantu aplikasi wireshark.



Gambar 4.5 Arsitektur Forensik jaringan

Setelah data tersebut terekam maka proses analisis dilakukan oleh peneliti dalam menguraikan karakteristik file log flooding yang telah terdeteksi. Pada gambar 4.6 Peneliti bertindak sebagai investigator forensic jaringan dimulai dari mempersiapkan sistem, simulasi serangan untuk mendeteksi pada jaringan, dan menganalisis karakterinstik file log.

Pengambilan log dilakukan secara real time, kemudian dikumpulkan file log tersebut agar bias dianalisis oleh peneliti investigator. Dengan menggunaan model proses forensic, sebuah file log seharusnya bisa menjawab pertanyaan penyerangan apa yang terjadi, siapa yang menyerang dari IP addressnya, kapan terjadi penyerangan, di prangkat manakah telah terjadi penyerangan, bagaimana itu bisa terjadi dan mengapa itu bisa terjadi.

4.4.2 Analisis Model Proses Forensik

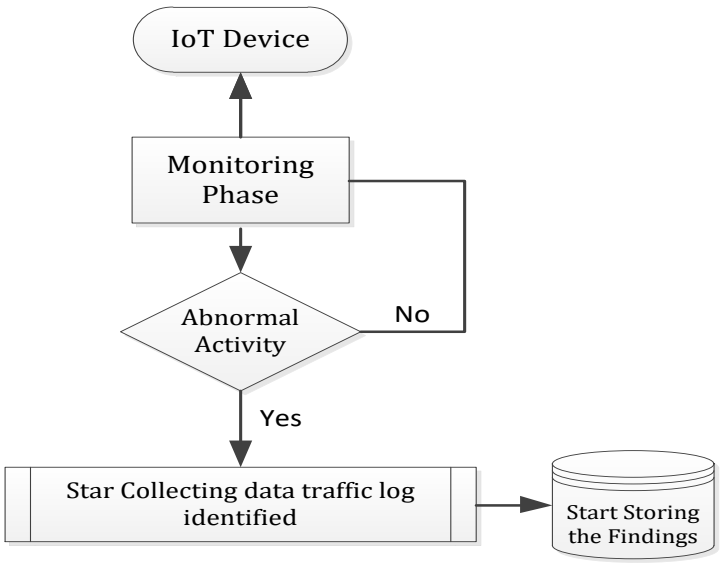
Level forensik pada pemeriksaan perangkat, forensik jaringan digunakan untuk menganalisis dan mencatat lalu lintas. Jumlah perangkat *Internet of Things* (IoT) akan menghasilkan data yang sangat besar. Jumlah bukti data akan sangat besar dan akan sangat sulit untuk menganalisis data dan sulit untuk mengidentifikasi bukti yang digunakan untuk mengidentifikasi dalam menemukan serangan *flooding* dan pemantauan sehingga dapat mengidentifikasi sumber serangan pada perangkat yang terinfeksi. Hasil analisis ini memiliki sembilan tahap Model Proses Forensik :

a. Tahap *Preparation and Authorization*

Pada tahap ini, forensik jaringan hanya berlaku untuk lingkungan di mana alat keamanan jaringan seperti penganalisis paket, perangkat lunak untuk pengukuran arus lalu lintas digunakan di berbagai titik strategis di jaringan untuk mendeteksi serangan flooding pada perangkat *Internet of Things* (IoT). Staf yang menangani alat-alat ini harus dilatih untuk memastikan bahwa bukti maksimal dan kualitas dapat dikumpulkan untuk memfasilitasi atribusi kejahatan. Otorisasi yang diperlukan untuk memantau lalu lintas jaringan diperoleh dan kebijakan keamanan yang ditetapkan dengan baik di tempat sehingga privasi individu dan organisasi tidak dilanggar.

b.* Tahap *Detection of Incident / Crime

Mendeteksi peringatan yang dihasilkan oleh berbagai alat keamanan, menunjukkan pelanggaran keamanan atau pelanggaran kebijakan, diamati. Setiap kejadian yang tidak sah dan aktivitas abnormal yang diperhatikan akan dianalisis. Konfirmasi insiden menghasilkan dua arah yaitu respons insiden dan pengumpulan data.



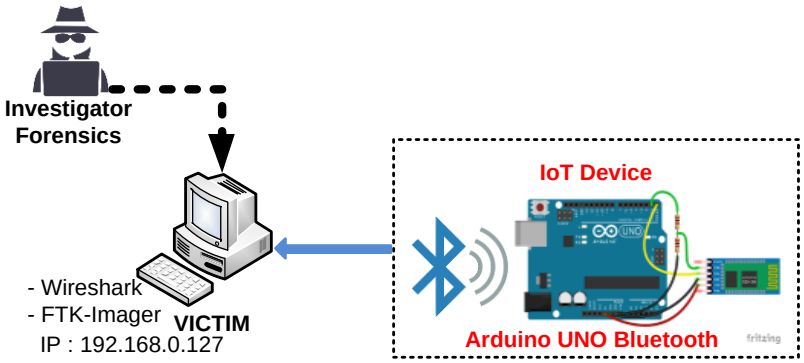
Gambar 4.6 *Detection of Findings Log*

c. Tahap *Incident Response*

Tanggapan terhadap kejahatan atau intrusi yang terdeteksi dimulai berdasarkan informasi yang dikumpulkan untuk memvalidasi dan menilai insiden tersebut. Tanggapan yang dimulai tergantung pada jenis serangan yang diidentifikasi dan dipandu oleh kebijakan organisasi, hukum dan bisnis. Fase ini hanya berlaku untuk kasus di mana penyelidikan dimulai ketika serangan sedang berlangsung dan bukan kriminis notitia (setelah pemberitahuan kejahatan).

d. Tahap *Collection of Network Traces*

Pengkoleksian barang bukti dalam penelitian ini menggunakan rekaman jaringan traffic log pada perangkat *Internet of Things* (IoT). Proses mengambil payload sebagai file serangan flooding dalam penelitian ini sebagaimana terdapat pada gambar 4.7 sebagai berikut :



Gambar 4.7 Data Collection Stage

e. Tahap *Protection and Preservation*

Data asli yang diperoleh dalam bentuk traffic log disimpan di dalam perangkat. Hash dari semua data jejak diambil dan data dilindungi. Salinan lain dari data akan digunakan untuk analisis dan lalu lintas jaringan yang dikumpulkan asli dipertahankan. Pada tahap ini akan menggunakan aplikasi FTK Imager untuk membuat hash data.

```
Created By AccessData® FTK® Imager 3.2.0.0

Case Information:
Acquired using: ADI3.2.0.0
Case Number: 02
Evidence Number: 02
Unique Description: Forensik IoT Device
Examiner: Randi Rizal
Notes: Hash Device

---------------------------------------------------------------

Information for D:\Tesis-Randi\IoT Hash.ad1:
[Custom Content Sources]
 Documents:C:\Users\user\Documents|coba|bluetooth-traffic-capture-port 137.pcapng(Exact)
[Computed Hashes]
 MD5 checksum:    ad9296005344b1dad3aab84b117c522e
 SHA1 checksum:   32cb7fb94dd2f9fe11766eae67a624a62cca7012

Image information:
 Acquisition started:   Wed Jun 06 10:43:57 2018
 Acquisition finished:  Wed Jun 06 10:43:58 2018
 Segment list:
  D:\Tesis-Randi\IoT Hash.ad1

Image Verification Results:
 Verification started:  Wed Jun 06 10:43:58 2018
 Verification finished: Wed Jun 06 10:43:58 2018
 MD5 checksum:    ad9296005344b1dad3aab84b117c522e : verified
 SHA1 checksum:   32cb7fb94dd2f9fe11766eae67a624a62cca7012 : verified
```

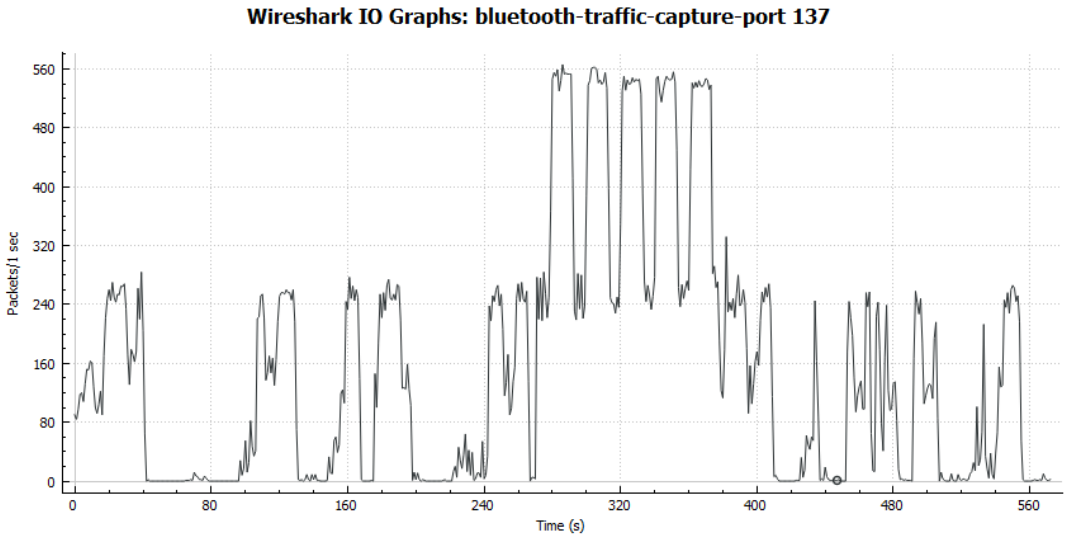
Gambar 4.8 Hash Evidence of Log Traffic

f. Tahap *Examination*

Peneliti forensik dalam memeriksa file log yang ditemukan di log lalu-lintas bluetooth dalam pengambilan (p.cap) dengan memasukkan parameter yang akan dipasang. Proses pemeriksaan akan menangkap lalu lintas jaringan pada perangkat IoT dengan menggunakan aplikasi wireshark.

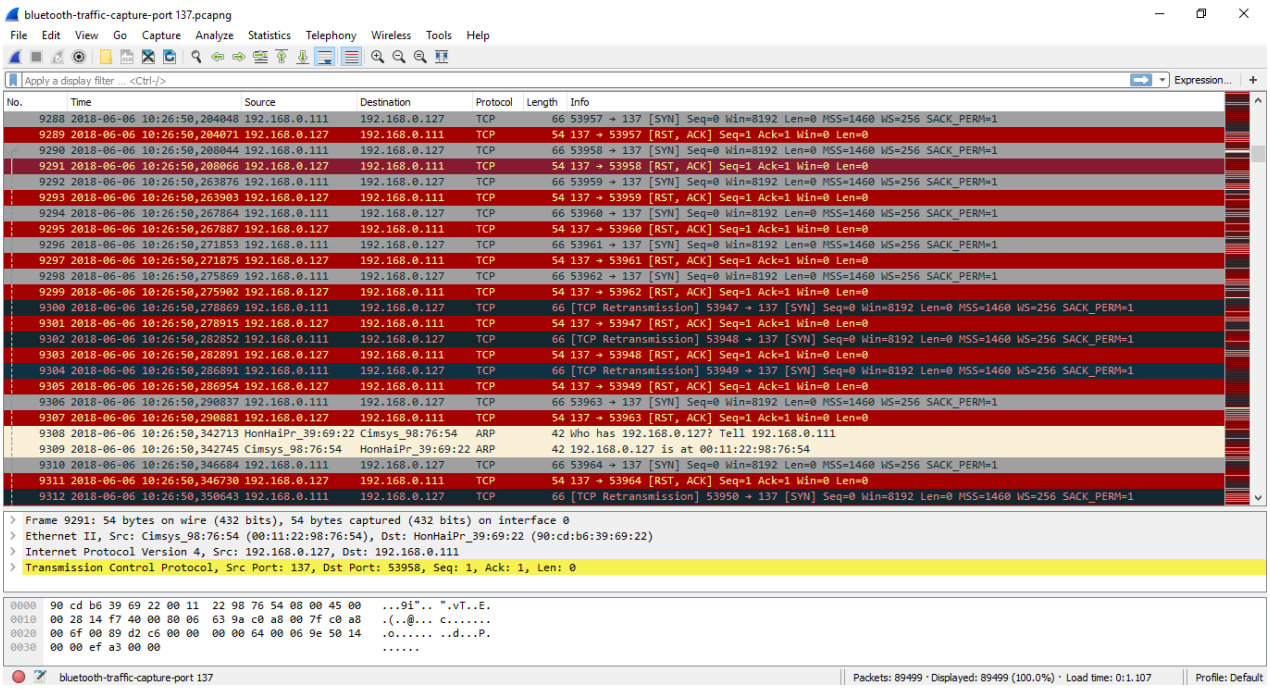
g. Tahap *Analysis*

Pada tahap ini analisis file log akan diperiksa, file-file log yang telah dipulihkan akan diperiksa satu per satu untuk menentukan perubahan dalam jaringan dan untuk melihat *time stamp*. Serangan *flooding* akan terlihat ketika permintaan ke perangkat IoT meningkatkan trafik penangkapan yang merupakan anomali. Kemudian serangan flooding dikirim dari penyerang sehingga lalu lintas akan meningkat. Selain lalu lintas yang dilakukan penyidik menggunakan aplikasi wireshark untuk menangkap lalu lintas, juga dapat dilihat peningkatan traffic jaringan seperti pada gambar 4.9.



Gambar 4.9 *IO Graph Traffic Log*

Setelah file log dicatat, file log akan diambil dan dianalisis menggunakan Wireshark untuk mendapatkan bukti forensik ini. Dalam gambar terlihat total paket yang tercatat melebihi 89000 paket dalam interval waktu 8 menit. Seperti yang ditunjukkan pada gambar 4.10.

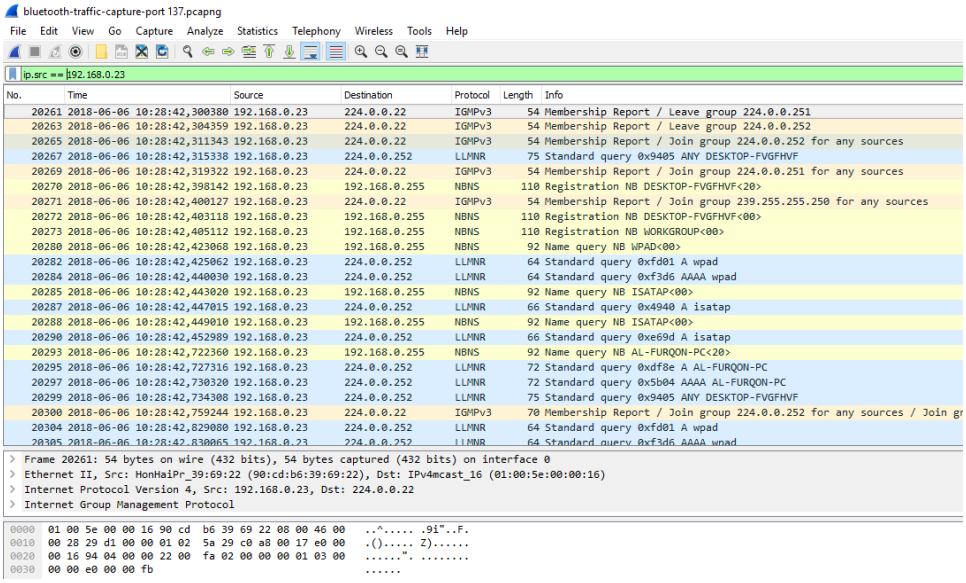


Gambar 4.10 *Traffic Log in Wireshark*

h. Tahap *Investigation and Attribution*

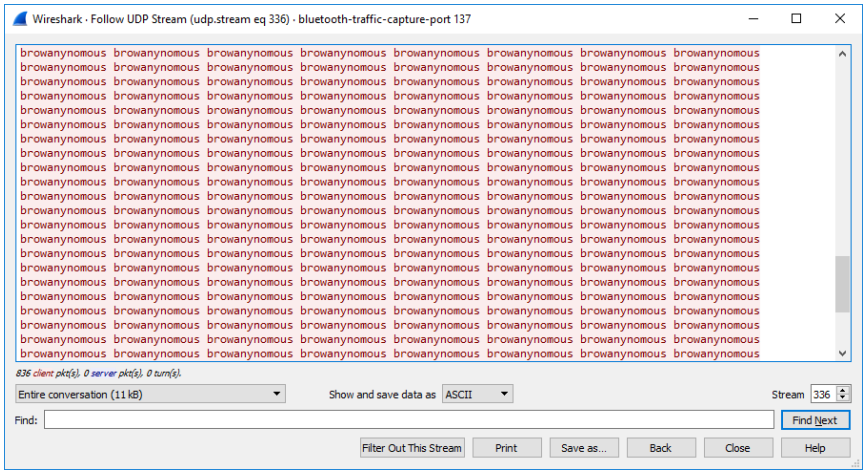
Informasi yang diperoleh dari jejak bukti digital digunakan untuk mengidentifikasi insiden tersebut. Ini akan membantu dalam pelacakan jejak sumber, rekonstruksi skenario serangan dan atribusi ke sumber.

Setelah melihat *trrafic* yang masuk maka dapat diambil *file log* secara *real team* sehingga dapat dianalisis menggunakan Wireshark dalam mencari tau penyerang yang mengirimkan paket berlebihan ke perangkat IoT. Dalam proses analisis aktivitas illegal didalam jaringan, Wireshark mampu melihat atau menganalisis paket secara offline seperti pada gambar 4.11 dengan menggunakan bantuan filter berdasarkan IP address (ip.scr). filter yang dilakukan terhadap ip penyerang sebagai berikut :



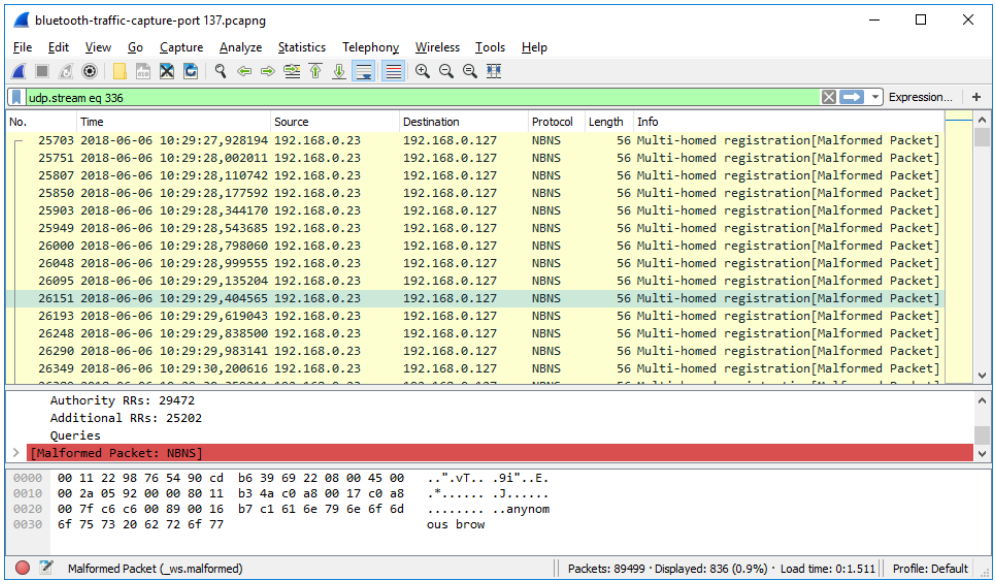
Gambar 4.11 *Filter ip.src*

IP address 192.168.0.23 melakukan serangan *flooding* kemudian dapat menganalisis dengan filter ip.src==192.168.0.23, pilih salah satu baris untuk melakukan analisis, kemudian klik kanan Follow UDP Stream pada gambar 4.12 kemudian *clouse*.



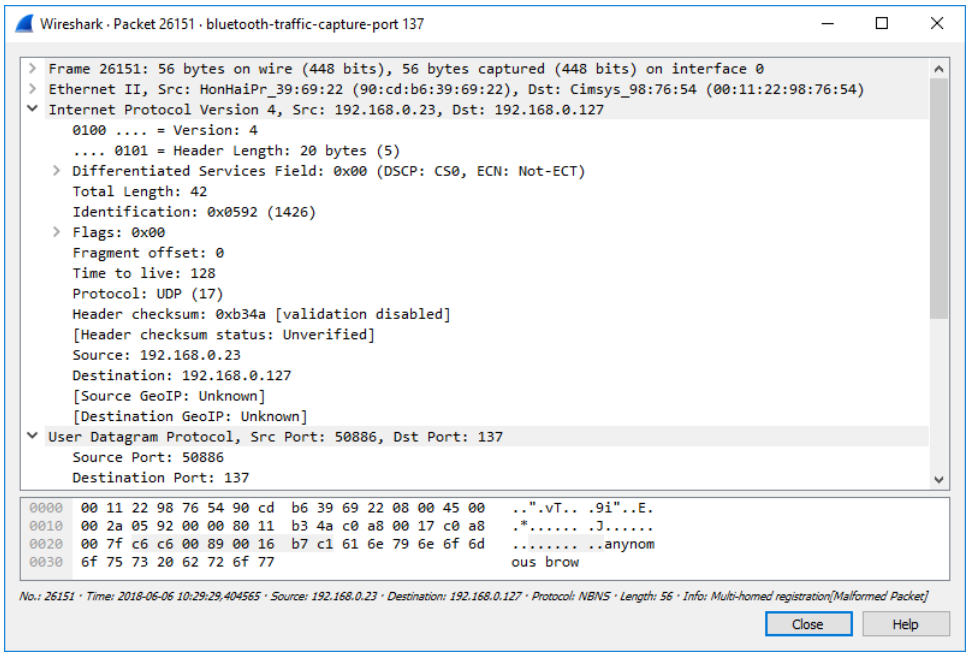
Gambar 4.12 *Follow UDP*

Setelah memlilih *clouse* maka akan muncul gambar 4.13 dibawah ini:



Gambar 4.13 Hasil follow UDP

Dari kumpulan baris tersebut dapat dipilih salah satu baris untuk melakukan analisis pada setiap bagian *frame* yang mewakili sebuah *packet frame* pada serangan *flooding* dari IP address 192.168.0.23 memiliki panjang (length) berkisar pada angka 50-an Bytes (54 Bytes). Pada bagian Internet Protocol Version 4, source IP terbaca sebagai 192.168.0.23 dan destination IP address terbaca 192.168.0.127 dengan *header length* 20 Bytes dan total *length* 42. Pada bagian User Datagram Protocol, source port terbaca sebagai 50886 dan destination port terbaca sebagai 137. Apabila filter dikembalikan pada ip.src== 192.168.0.23 dan diteliti pada frame yang lain, source port-nya berubah-rubah, namun tetap dalam rentang yang besar (port 51000-60000). Hasil capture frame analisis IP address 192.168.0.23 pada gambar 4.14.



Gambar 4.14 Hasil frame

Selanjutnya dilakukan analisis IP address penyerang lainnya seperti langkah-langkah diatas maka dapat dikumpulkan data analisis untuk barang bukti terdapat IP address penyerang yang melakukan serangan *flooding* terhadap perangkat *Internet of Things (IoT)*. Berikut adalah IP address yang terdeteksi melakukan serangan dengan klasifikasi beban paket dan kecepatan yang berbeda setiap bytenya :

No.	Alamat IP	Packets	Bytes	Tx Packets	Tx Bytes	Rx Packets	Rx Bytes	Klasifikasi
1	192.168.0.15	4.289	263k	2.260	153k	2.029	109k	MEDIUM
2	192.168.0.23	47.027	2660k	44.520	2524k	2.507	135k	HIGH
3	192.168.0.25	6.319	384k	3.278	220k	3.041	163k	MEDIUM
4	192.168.0.71	7.008	424k	3.629	242k	3.379	182k	MEDIUM
5	192.168.0.111	5.210	317k	2.724	183k	2.486	134k	MEDIUM
6	192.168.0.128	3.100	192k	1.668	115k	1.432	77k	MEDIUM
7	192.168.0.132	3.445	212k	1.838	126k	1.607	86k	MEDIUM
8	192.168.0.221	7.448	442k	3.745	242k	3.703	199k	MEDIUM
9	192.168.0.237	3.908	218k	3.908	218k	0	0	MEDIUM
10	192.168.0.255	796	75k	0	0	796	75k	LOW

Tabel. 4.1 Klasifikasi Paket Serangan

Selain itu, analisis dilanjutkan dengan titik akhir modul statistik di Wireshark yang digunakan untuk mengumpulkan paket serangan yang terdapat dalam file log selama proses simulasi serangan pada perangkat Internet of Things (IoT). Pada tabel 4.1 di atas menjelaskan bahwa alamat IP yang terdeteksi sebagai penyerang memiliki beban yang berbeda pada setiap paket dan pada kecepatan yang berbeda di setiap bytenya.

i. Tahap *Presentation*

Pada tahap presentasi adalah tahap terakhir dalam model proses forensik generik. Tahap ini adalah presentasi dari semua temuan dalam penelitian ini. Berdasarkan analisis yang telah dilakukan maka diperoleh alamat IP address penyerang dengan IP 192.168.0.23 mempunyai jumlah paket yang sangat besar yaitu 47.027 (2660k) dengan beban transaksi paket sebesar 44.520 (2524) yang menjadi temuan dalam skenario penelitian mendeteksi serangan flooding pada perangkat *Internet of Things (IoT)*.

Bab 5 Kesimpulan dan Saran

Pada bagian ini menjelaskan kesimpulan dari hasil penelitian yang telah dilakukan berdasarkan tujuan dan perumusan masalah penelitian, yaitu : 1) menguji framework *Generic Network Forensics Process Model* untuk mendeteksi serangan *flooding*. 2) melakukan pengujian untuk mendeteksi kinerja serangan *flooding* pada perangkat *Internet of Things (IoT).*

5.1 Kesimpulan

Kesimpulan yang telah didapatkan selama proses penelitian dalam melakukan deteksi serangan flooding pada perangkat *Internet of Things* (IoT) menyimpulkan bahwa :

1. Pengujian framework *Generic Network Forensics Process Model* untuk mendeteksi serangan *flooding* pada perangkat *Internet of Things* (IoT) dapat digunakan untuk membantu proses investigasi forensik digital. Dan menyajikan bukti digital yang legal di dalam persidangan.
2. Kinerja serangan *flooding* pada perangkat *Internet of Things* (IoT) dengan *log file* yang diambil untuk dianalisis aktivitas tindakan illegal yang terjadi pada lingkungan perangkat *Internet of Things* (IoT). Berdasarkan analisis file log, terdapat sebanyak 10 IP *address* penyerang yang melakukan tindakan illegal dengan frekuensi penerimaan data. Hal tersebut menunjukkan bahwa ada serangan *flooding*, serangan yang mengirimkan banyak data pada target sehingga sumber daya yang dimiliki mengalami gangguan atau pada ahirnya dapat rusak *device*.

5.2 Saran

Saran yang dapat diberikan berdasarkan kesimpulan penelitian diatas adalah :

1. Saran yang dapat diberikan untuk keperluan penelitian selanjutnya adalah pengembangan dari model proses forensik untuk kasus *Internet of Things* (IoT) secara kompleks dan menyeluruh.
2. Melakukan pengembangan unuk melakukan implementasi pada sistem jaringan workstation yang lebih besar seperti *smart city, smart detector* dll.

Daftar Pustaka

Atamli, A.W. & Martin, A., 2014. Threat-Based Security Analysis for the Internet of Things. *2014 International Workshop on Secure Internet of Things*, pp.35–43. Available at: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm? arnumber=7058906.

Bandyopadhyay, Debasis, and Jaydip Sen. "Internet of things: Applications and challenges in technology and standardization." *Wireless Personal Communications* 58.1 (2011): 49-69.

Borgohain, T., Kumar, U. & Sanyal, S., 2015. Survey of Security and Privacy Issues of Internet of Things. *arXiv preprint arXiv:1501.02211*, p.7. Available at: http://arxiv.org/abs/1501.02211.

Buric, J., and D. Delija. "Challenges in Network forensics." *Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2015 38th International Convention on*. IEEE, 2015.

Charlie Osborne, 2016. Insulin pump vulnerabilities could lead to overdose | ZDNet. Available at: http://www.zdnet.com/article/insulin pumpvulnerabilities-could-lead-to-overdose/.

E.S. Pilli, R.C. Joshi, & R. Niyogi**. "**A Generic Framework for Network Forensics". *International Journal of Computer Applications (IJCA) (0975 – 8887) Volume 1 – No. 11*, 2012.

Hachem, S., Teixeira, T. & Issarny, V., 2012. Ontologies for the internet of things. *Proceedings of the 8th Middleware Doctoral Symposium on - MDS '11*, pp.1–6.

Huuck, R., 2015. IoT: The Internet of Threats and Static Program Analysis Defense. *EmbeddedWorld 2015: Exibition & Conferences*, p.493.

Hossain, Md Mahmud, Maziar Fotouhi, and Ragib Hasan. "Towards an Analysis of Security Issues, Challenges, and Open Problems in the Internet of Things." *Services (SERVICES), 2015 IEEE World Congress on*. IEEE, 2015.

Iswardani, A., and Riadi, I. "Denial Of Service Log Analysis Using Density K-Means Method," vol. 83, no. 2, pp. 299–302, 2016.

L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," Computer networks, vol. 54, no. 15, pp. 2787–2805, 2013.

Mualfah D. and Riadi I. "Network Forensics For Detecting Flooding Attack On Web Server" *(IJCSIS) International Journal of Computer Science and Information Security, Vol.15, 2017.*

Nguyen, K., Tran, D., Ma, & Shama, D. (2014) An Approach to Detect Network Attacks Applied for Network Forensics, 655-660.

Oriwoh, Edewede, et al. "Internet of Things Forensics: Challenges and approaches." *Collaborative Computing: Networking, Applications and Worksharing (Collaboratecom), 2013 9th International Conference Conference on*. IEEE, 2013.

Oriwoh, Edewede, and Paul Sant. "The Forensics Edge Management System: A Concept and Design." *Ubiquitous Intelligence and Computing, 2013 IEEE 10th International Conference on and 10th International Conference on Autonomic and Trusted Computing (UIC/ATC)*. IEEE, 2013.

Ramjee Prasad, Antonietta Stango, Neeli Prasad & Sachin Babar."Proposed Embedded Security Framework for Internet of Things (IoT)**".** https://www.researchgate.net/publication/252013823, 2011.

www.gartner.com, "Gartner Says the Internet of Things Will Transform the Data Center," http://www.gartner.com/newsroom/id/2684616, 2014.

www.idc.com, "Finding Success in the New IoT Ecosystem: Market to Reach $3.04 Trillion and 30 Billion Connected "Things" in 2020, IDC Says ," http://www.idc.com/getdoc.jsp?container Id=prUS25237214, 2014.

Walsh Ray, 2016. IoT Botnet Launching Massive DdoS Attacks on Websites - BestVPN.com. Available at: https://www.bestvpn.com/iot-botnet-mirai-ddos/[Accessed December 14, 2016].

Y. Huang and G. Li, "A semantic analysis for internet of things," in Intelligent Computation Technology and Automation (ICICTA), 2010 International Conference on, vol. 1. IEEE, 2010, pp. 336–339.

Zawoad, Shams, and Ragib Hasan. "FAIoT: Towards Building a Forensics Aware Eco System for the Internet of Things." *Services Computing (SCC), 2015 IEEE International Conference on*. IEEE, 2015.

LAMPIRAN

Source Code Bluetooth Arduino

#Mengaktifkan Bluetooth

```
if (!mBluetoothAdapterisEnabled()) {
            Intent enableBtIntent = new Intent
          (BluetoothAdapter.ACTION_REQUEST_ENABLE);
   startActivityForResult (enableBtIntent, REQUEST_ENABLE_BT);
}
```

#Mencari Bluetooth Device

```
Set pairedDevices = mBluetoothAdapter.getBondedDevices();
// If there are paired devices
if (pairedDevices.size() &gt; 0) {
   // Loop through paired devices
   for (BluetoothDevice device : pairedDevices) {
      // Add the name and address to an array adapter to show in a ListView
      mArrayAdapter.add(device.getName() + "\n" + device.getAddress());
   }
}
```

#Menghubungkan Bluetooth Device

1. Menggunakan kelas *BluetoothDevice*, dapatkan *BluetoothSocket* dengan memanggil *createRfcommSocketToServiceRecord(UUID)*.
2. Memulai koneksi dengan memanggil method *connect()*.

```
private class ConnectThread extends Thread {
    private final BluetoothSocket mmSocket;
    private final BluetoothDevice mmDevice;
    public ConnectThread(BluetoothDevice device) {
        // Use a temporary object that is later assigned to mmSocket,
        // because mmSocket is final
        BluetoothSocket tmp = null;
        mmDevice = device;
         // Get a BluetoothSocket to connect with the given BluetoothDevice
        try {
            // MY_UUID is the app's UUID string, also used by the server code
            tmp = device.createRfcommSocketToServiceRecord(MY_UUID);
        } catch (IOException e) { }
        mmSocket = tmp;
    }
    public void run() {
        // Cancel discovery because it will slow down the connection
        mBluetoothAdapter.cancelDiscovery();
        try {
            // Connect the device through the socket. This will block
            // until it succeeds or throws an exception
            mmSocket.connect();
        } catch (IOException connectException) {
            // Unable to connect; close the socket and get out
            try {
                mmSocket.close();
            } catch (IOException closeException) { }
```

```
                return;
            }
        // Do work to manage the connection (in a separate thread)
        manageConnectedSocket(mmSocket);
    }
    /** Will cancel an in-progress connection, and close the socket */
    public void cancel() {
        try {
            mmSocket.close();
        } catch (IOException e) { }
    }
}
```

#Memanage koneksi yang sudah terbentuk

```
  private class ConnectedThread extends Thread {
      private final BluetoothSocket mmSocket;
      private final InputStream mmInStream;
      private final OutputStream mmOutStream;
      public ConnectedThread(BluetoothSocket socket) {
          mmSocket = socket;
          InputStream tmpIn = null;
          OutputStream tmpOut = null;
          // Get the input and output streams, using temp objects because
          // member streams are final
          try {
              tmpIn = socket.getInputStream();
              tmpOut = socket.getOutputStream();
          } catch (IOException e) { }
          mmInStream = tmpIn;
          mmOutStream = tmpOut;
      }
      public void run() {
          byte[] buffer = new byte[1024];  // buffer store for the stream
          int bytes; // bytes returned from read()
          // Keep listening to the InputStream until an exception occurs
          while (true) {
              try {
                  bytes = mmInStream.read(buffer);
                  // Send the obtained bytes to the UI activity
                  mHandler.obtainMessage(MESSAGE_READ, bytes, -1, buffer)
                          .sendToTarget();
              } catch (IOException e) {
                  break;
              }
          }
      }
      /* Call this from the main activity to send data to the remote device */
      public void write(byte[] bytes) {
          try {
              mmOutStream.write(bytes);
          } catch (IOException e) { }
      }
```