Implementasi dan Analisa *Attack Tree* pada *Vulnerable Machine* Toppo: 1 Berdasarkan *Time Metric, Cost Metric,* dan *Frequency Metric*

1st Irgi Fahrezi Salim
Fakultas Rekayasa Industri
Universitas Telkom
Bandung, Indonesia
irgifahrezisalim@student.telkomuniversity.ac.id

2nd Adityas Widjajarto
Fakultas Rekayasa Industri
Universitas Telkom
Bandung, Indonesia
adtwjrt@telkomuniversity.ac.id

3rd Ahmad Almaarif
Fakultas Rekayasa Industri
Universitas Telkom
Bandung, Indonesia
ahmadalmaarif@telkomuniversity.ac.id

Abstrak—Penelitian ini melakukan implementasi dan analisa *attack tree* pada *vulnerable machine* Toppo: 1 berdasarkan *time* metric, *cost* metric, dan *frequency* metric. Nilai time, cost, dan frequency didapatkan setelah melakukan eksploitasi dengan mencoba *walkthrough* hingga memperoleh *privileged environment access* dari VM Toppo: 1. Skenario eksploitasi yang terdapat pada walkthrough memiliki tahapan yaitu, information *gathering*, *scanning*, *gaining access*, exploit, dan *privileged escalation*. *Walkthrough* tersebut dapat digambarkan dengan activity diagram. Analisa dan perumusan *attack tree* disusun dengan menggunakan pendekatan CubeSat *Security Attack Tree Analysis* dan SAND *gate*. *Attack tree* tersebut mewakili berbagai tahapan eksploitasi dan dapat dilakukan pemeringkatan berdasarkan metrics. Metrics tersebut dilakukan analisa untuk mendapatkan pemeringkatan *attack tree*. *Attack tree* 1 merupakan jalur tercepat jika dibanding dengan attack tree lainnya dengan nilai real time 167,868 detik. Data yang didapatkan berdasarkan cost metric, attack tree 1 memiliki nilai cost yang rendah jika dibanding dengan attack tree lainnya dengan nilai cost 21. Data yang didapatkan berdasarkan frequency metric, attack tools Dirb dan Arp-scan memiliki persentase terbesar dalam penggunaannya pada lima *attack tree* sebesar 40% dan 30%. Kelanjutan dari penelitian ini dapat dilakukan dengan menambahkan faktor kerentanan dengan melakukan vulnerability scanning pada sistem.

Kata kunci—*attack tree, time, cost, frequency*, toppo: 1

I. PENDAHULUAN

Keamanan informasi adalah bagian terpenting dari pengelolaan organisasi, aktivitas TI dapat terganggu ketika masalah timbul pada keamanan informasi terkait kerahasiaan, integritas, dan ketersediaan. Keamanan informasi merupakan upaya untuk melindungi sumber daya TI yang dimiliki. Tujuan dari keamanan informasi ini adalah untuk memastikan kelangsungan sistem dan mitigasi potensi risiko kerusakan pada sistem [1]. Jika terdapat celah keamanan pada informasi maka, penyerang akan dengan mudah mengambil alih sistem dan data yang telah dibangun oleh organisasi.

Penyerang atau *hacker* merupakan seseorang atau sekumpulan orang yang melakukan kegiatan pembobolan sistem. Jika aktivitas berlangsung di jaringan komputer, seorang peretas dapat memperoleh akses ke jaringan dan mendapatkan akses ke semua pengguna sistem yang terhubung ke jaringan [2]. Seiring dengan perkembangan teknologi maka para penyerang memiliki banyak cara untuk melakukan penyusupan pada suatu sistem ataupun peretasan kelemahan pada web *server* untuk keuntungan pribadi maupun organisasi. Salah satu cara untuk mengetahui celah keamanan pada *vulnerable machine* dengan waktu dan biaya yang sedikit yaitu, dapat disusun menggunakan *attack tree.*

Attack tree adalah sebuah diagram atau model matematis berstruktur *tree* yang mewakili sistem yang ingin diambil alih oleh penyerang. Model *attack tree* ini menggambarkan pilihan atau tujuan yang dilakukan penyerang. *Attack tree* ini terdiri dari, simpul atas yang mewakili tujuan akhir penyerang dan simpul paling bawah menggambarkan operasi yang dilakukan penyerang untuk mengeksploitasi kerentanan dalam pertahanan sistem [3]. *Attack tree* memiliki bagian yang dapat dilakukan perhitungan nilainya, bagian tersebut dinamakan *metric. Metrics* memiliki beberapa perhitungan *metrics* yaitu, *time, cost, frequency, probability to successfully attack, skill level needed for attacks, special equipment needed for attacks,* dan *combinations of the above metrics* [4]. Hasil perhitungan *metrics* tersebut dapat digunakan sebagai landasan untuk pemeringkatan *attack tree*.

Pada penelitian ini berfokus pada "Implementasi Dan Analisa *Attack Tree* Pada *Vulnerable Machine* Toppo: 1 Berdasarkan *Time Metric* Dan *Cost Metric"*. Bagi pengelola server, *attack tree* dapat digunakan sebagai dasar penguatan keamanan sistem dengan menganalisa cara-cara di mana sistem itu diserang. Sedangkan bagi penyerang, *attack tree* hasil akhirnya dapat digunakan untuk mengetahui jalur tercepat dan dengan *cost* rendah berdasarkan analisa *metric.* Toppo: 1 merupakan sebuah *vulnerable machine operating system* yang sederhana. *Vulnerable machine* Toppo: 1 merupakan serangkaian sistem operasi yang dibuat untuk melatih, menambah wawasan, dan pengetahuan mengenai penetrasi. Toppo: 1 dikembangkan oleh Hadi Mene pada 12 Juli 2018 pada *website* Vulnhub.

II. KAJIAN TEORI

A. Keamanan Sistem Informasi

Keamanan sistem informasi merupakan upaya untuk melindungi aset informasi dari potensi ancaman. Secara tidak langsung, keamanan informasi menjamin kelangsungan bisnis, mengurangi risiko ancaman, dan mengoptimalkan pengembalian modal yang diinvestasikan [5]. Keamanan

informasi ini memiliki tujuan untuk mencegah terjadinya ancaman pada suatu sistem dan melakukan pendeteksian serta perbaikan kerusakan pada sistem.

B. Eksploitasi

Eksploitasi merupakan sebuah jenis program yang dirancang untuk menargetkan kelemahan tertentu pada sistem dikenal sebagai celah keamanan pada perangkat keras atau perangkat lunak [6].

C. *Threat*

Threat atau ancaman merupakan sebuah kemungkinan terjadinya gangguan yang muncul terhadap sistem maupun jaringan dan akan mengakibatkan bahaya pada keberlangsungan suatu sistem. Ancaman sistem dapat dibagi menjadi dua jenis: aktif dan pasif. Ancaman aktif termasuk penipuan dan kejahatan dunia maya, dan ancaman pasif termasuk bencana alam, kesalahan manusia, dan kegagalan sistem atau lingkungan [7].

D. Lubuntu

Lubuntu merupakan sebuah sistem operasi *open-source software* yang menggunakan lingkungan *desktop environment*. Lubuntu ini merupakan varian dari Ubuntu yang memiliki sifat DE (*desktop environment)*. Lubuntu ini digunakan sebagai *penyerang* dalam eksperimen.

E. *Vulnerable Machine* Toppo: 1

Salah satu *vulnerable machine operating system* yang sederhana yaitu, Toppo: 1. Toppo: 1 ini dibuat oleh Hadi Mene pada 12 Juli 2018 pada *website* Vulnhub, dengan tujuan untuk memperbanyak pengalaman dan wawasan di dalam dunia *penetration testing*. Toppo: 1 ini merupakan edisi pertama yang telah dibuat oleh Hadi Mene.

F. Eksperimen

Eksperimen adalah sebuah metode penyelidikan pada variabel yang datanya belum ada, sehingga perlu dimanipulasi dengan pemberian *treatment* atau perlakuan tertentu kepada subjek, yang efeknya kemudian diamati atau diukur (data yang akan datang).

G. *Walkthrough*

Walkthrough merupakan sebuah cara untuk berinteraksi langsung secara antarmuka pengguna aplikasi dan melakukan pemeriksaan mekanisme teknologi dan referensi budaya untuk memahami alur pengguna dan membentuk pengalaman. Inti dari *walkthrough* ini adalah pengamatan tahapan demi tahapan dan dokumentasi layar, fungsi, dan alur kerja aplikasi secara teknis. Secara teknis pada *walkthrough* itu berarti *walkthrough* menyebutkan daftar software serta tools yang digunakan untuk melakukan penyerangan ke sistem. Tujuan dari *walkthrough* ini adalah agar pengguna mendapatkan kesempatan untuk membiasakan diri dengan sistem sebelum melakukan aktivitas tes ke sistem secara langsung [8].

H. *Activity Diagram*

Activity diagram ini dapat dimodelkan sebagai aliran kerja dari aktivitas satu ke aktivitas lainnya atau dari satu aktivitas ke dalam keadaan sesaat. *Activity diagram* ini juga bermanfaat untuk menggambarkan atau mendeskripsikan perilaku paralel maupun menjelaskan bagaimana perilaku melakukan interaksi antar *use case* [9].

I. *Data Flow Diagram*

Data flow diagram merupakan metode analisis dan desain terstruktur. Ini merupakan indera visual untuk menggambarkan model logika dan mengekspresikan transformasi data pada sebuah sistem [10].

J. Pengukuran *Walkthrough*

Pengukuran adalah aktivitas menentukan nilai kuantitas tertentu. Pengukuran merupakan penentuan besaran, dimensi, atau kapasitas, biasanya mengacu pada standar atau satuan ukuran. Pengukuran juga dapat diartikan sebagai pemberian nilai numerik terhadap atribut atau ciri tertentu yang dimiliki oleh seseorang, benda atau objek menurut aturan atau rumusan yang jelas dan disepakati [11]. Pengukuran *walkthrough* terdiri dari 3 pengukuran yaitu, *time, cost,* dan *frequency* penggunaan *tools.*

K. *Attack Tree*

Attack tree merupakan sebuah diagram konseptual yang akan menunjukkan bagaimana aset atau target dapat diserang. *Attack tree* ini memiliki struktur hirarki dan representasi intuitif dari skenario serangan multi-langkah. Jadi, *attack tree* ini sangat praktis digunakan untuk menganalisis keamanan suatu sistem [4].

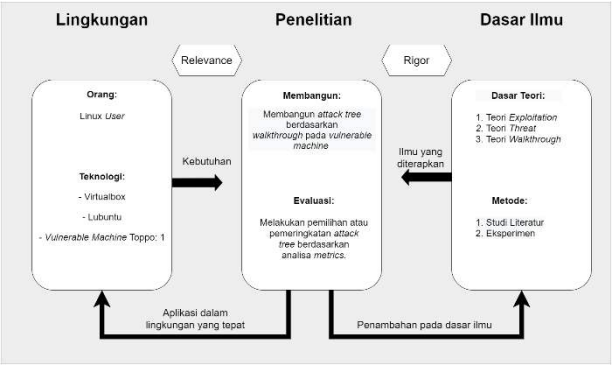
L. *Metrics*

Metrics merupakan sistem pengukuran dari *attack tree* yang dapat digunakan sebagai landasan membuat pemeringkatan pada *attack tree*. Terdapat 3 *metrics* yang digunakan yaitu, *time, cost,* dan, *frequency* [4].

III. METODE

A. Model Konseptual

Model Konseptual adalah sebuah model data awal yang diluaskan atau dikembangkan dengan cara mengidentifikasi, entitas, hubungan, kardinalitas, dan kendala dari domain masalah. Model konseptual dapat dimodifikasi atau diperbarui sebagai persyaratan penyimpanan data dari sistem yang sedang dalam pengembangan berubah [12]. Model konseptual pada penelitian ini digambarkan sebagai berikut:

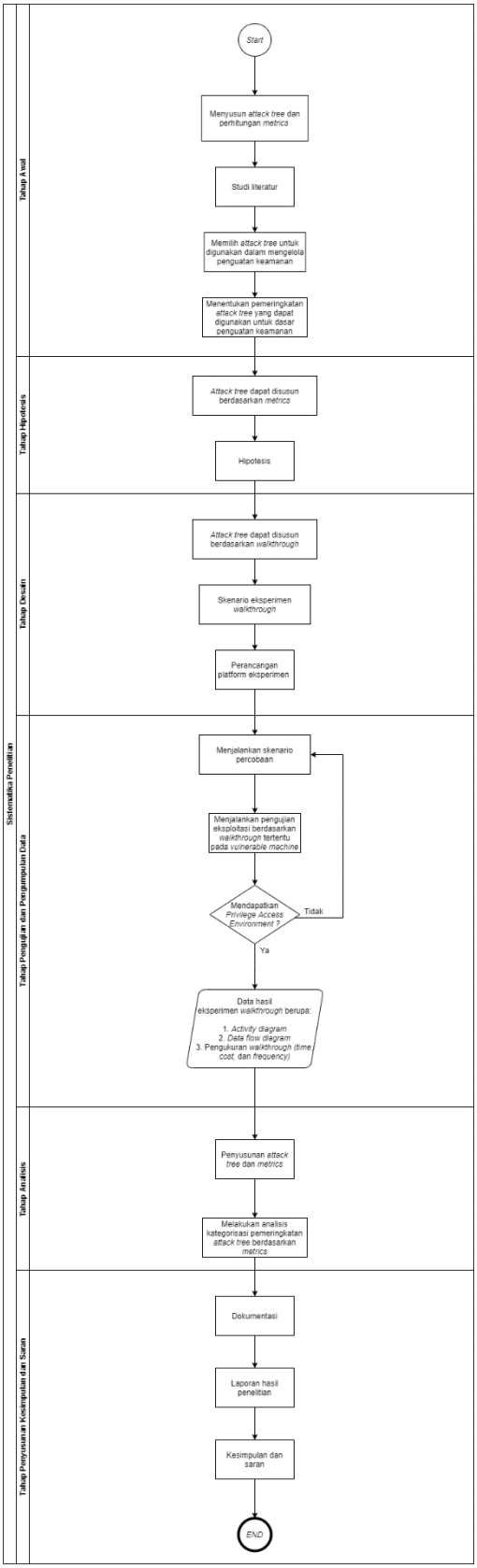


GAMBAR III.1
MODEL KONSEPTUAL

B. Sistematika Penyelesaian Masalah

Sistematika penelitian ini digunakan untuk menggambarkan tahapan-tahapan pada suatu penelitian yang

dijabarkan untuk memberikan penjelasan dan gambaran penelitian. Sistematika penelitian dilihat berikut.



GAMBAR III.2
SISTEMATIKA PENYELESAIAN MASALAH

IV. HASIL DAN PEMBAHASAN

Untuk dapat mencapai tujuan penelitian eksperimen berupa *penetration testing* berdasarkan *walkthrough*, diperlukan arsitektur berupa *hardware* (perangkat keras) dan *software* (perangkat lunak) untuk membantu menunjang pengumpulan data dari penelitian ini. Lingkup penelitian yang digunakan pada penelitian ini adalah Lubuntu dan *vulnerable machine* Toppo: 1 sebagai sarana eksperimen.

A. *Hardware and Software*

1. *Hardware*

Berikut ini merupakan rincian dari *hardware* yang digunakan:

TABEL IV.1
HARDWARE

Komponen	**Informasi**	
Spesifikasi Perangkat Keras: Asus A456U	*Processor*	*Intel* Core i7-7500U CPU @ 2.70GHz
	Memory	8GB RAM
	System Types	*64-bit operating system, x64-based processor*
	Operating Systems	*Windows 10 Home 64-bit (Main OS)*
	Hard Disk	1 TB

2. *Software*

Berikut ini merupakan rincian *software* apa saja yang digunakan:

TABEL IV.2
SOFTWARE

Type	***Software***	***Version***
Operating System	Lubuntu • *Memory:* 1024 MB • *Network: Bridged Adapter*	18.04
IT Asset	*Vulnerable Machine* Toppo • *Memory:* 1024 MB • *Network: Bridged Adapter*	1.0
Attack Tools	Nikto	2.1.5
	Nmap	7.60
	Dirb	2.22
	Netdiscover	0.3
	Arp-scan	1.9

Pada pengujian penelitian TA ini akan menggunakan *operating system, IT asset,* dan *attack tools*. Berikut ini merupakan penjelasan spesifikasi dari setiap *software* yang digunakan:

a. *Operating System*

1. Lubuntu

Lubuntu ini merupakan varian dari Ubuntu yang memiliki sifat DE (*desktop environment)*. Lubuntu ini digunakan sebagai *penyerang* dalam eksperimen.

b. *IT Asset*

2. Toppo: 1

Toppo: 1 merupakan sebuah *vulnerable machine operating system* sederhana yang sengaja dibuat oleh Hadi Mene dengan tujuan untuk memperbanyak pengalaman

dan wawasan didalam dunia *penetration testing*.

c. *Attack Tools*

3. Nikto

Nikto merupakan sebuah alat pemindai web *server open-source* (GPL) yang dapat memeriksa sebuah web dan akan mendapatkan laporan kerentanan dari web tersebut.

4. Nmap

Nmap merupakan *network mapper* yang bersifat *open-source.* Nmap ini berguna untuk pemindaian *port* pada sebuah *vulnerable machine.*

5. DIRB

DIRB merupakan sebuah *tool* yang bersifat CLI (*Command Line Interface)* yang dapat digunakan untuk mencari direktori yang ada pada sebuah *website*.

6. Netdiscover

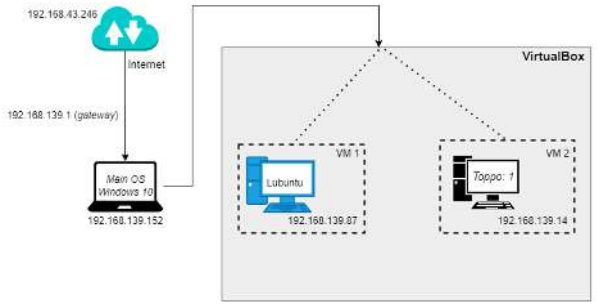
Netdiscover merupakan sebuah *tool* yang digunakan untuk mencari dan menemukan IP dalam proses *scanning*.

7. *Arp scan*

Arp scan merupakan *tool* CLI yang dapat digunakan untuk memeriksa dan melihat identitas *host* yang terhubung pada sebuah jaringan.

B. Platform Eksperimen

Berikut ini merupakan platform eksperimen pada penelitian ini.



GAMBAR IV.1
PLATFORM EKSPERIMEN

C. Daftar IP *Address*

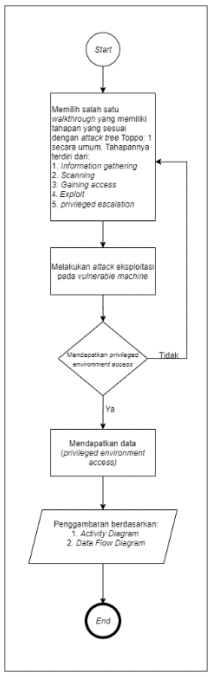
Pada Tabel IV.C di bawah ini, akan dijabarkan IP *address* yang akan digunakan pada penelitian ini.

TABEL IV.3
DAFTAR IP *ADDRESS*

Type	*Host*	*Default Gateway*	IP *Address*
Main OS	Windows 10	192.168.139.1/24	192.168.139.152/24
VM 1	Lubuntu		192.168.139.87/24
VM 2	Toppo: 1		192.168.139.14/24

D. Skenario Pengujian

Skenario pengujian ini menggunakan *data threat attack*. Di bawah ini merupakan perumusan *activity diagram* dan *data flow diagram* berdasarkan *walkthrough*:



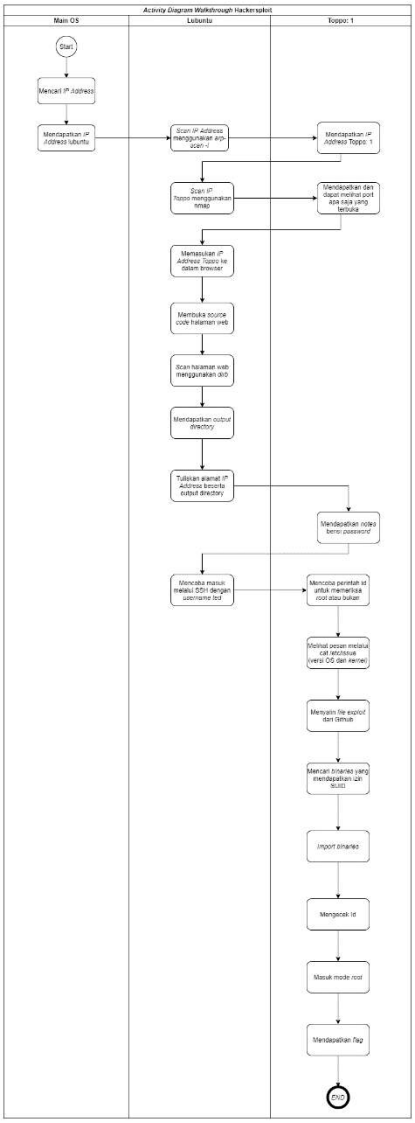
GAMBAR IV.2
PERUMUSAN *ACTIVITY DIAGRAM* DAN *DATA FLOW DIAGRAM* BERDASARKAN *WALKTHROUGH*

E. *Activity Diagram*

Activity diagram digunakan untuk menunjukkan tahapan tahapan menuju *privilege environment access.* Pada percobaan pertama ini akan dilakukan uji coba *walkthrough* Toppo: 1 yang berasal dari Hackersploit:

1. Tahap pertama dalam menjalankan sebuah *walkthrough* adalah mencari IP *address* pada *Main OS.*
2. Tahap kedua adalah mendapatkan IP *address* pada *Main OS.*
3. Selanjutnya, mencari IP *address* dari VM 2 menggunakan *command arp-scan -l.* Pastikan terlebih dahulu VM Toppo sudah dalam keadaan menyala.
4. Setelah itu, IP *address* ini digunakan untuk mengecek IP yang terdapat dalam suatu jaringan.
5. Lalu, lakukan *scan* IP tersebut menggunakan Nmap.
6. Selanjutnya, setelah mencoba *scan* IP menggunakan Nmap, Nmap akan memperlihatkan *port* TCP (*Transmission Control Protocol*) yang terbuka dan akan memperlihatkan spesifik dari *port-port* tersebut.
7. Setelah itu, tuliskan IP *address* yang telah didapatkan sebelumnya ke dalam *browser* dan akan ditampilkan halaman web Toppo.
8. Lalu, pada halaman utama Toppo buka *source code* dari halaman tersebut untuk mencari *clue* untuk tahap selanjutnya.
9. Selanjutnya, karena tidak mendapatkan *clue* dari *source code* tersebut, maka dicoba mencoba *command dirb* untuk mencari direktori Toppo.
10. Setelah mencoba *command* tersebut, dilakukan uji coba pada salah satu direktori yang tampil yaitu, */admin.*
11. Setelah itu, mencoba direktori yang didapatkan tersebut ke dalam *browser.*

12. Lalu, akan ditampilkan halaman yang berisikan *notes* dan didalam *notes* tersebut berisikan sebuah *password* untuk login ke dalam SSH.
13. Mencoba *login* melalui SSH dengan metode *hit and trial*, dengan cara mengasumsikan *password* sama dengan *username*.
14. Selanjutnya, setelah berhasil *login* melalui SSH cek terlebih dahulu menggunakan Id apakah login SSH ini sudah masuk ke dalam mode *root*.
15. Melihat pesan pada SSH dengan cara *cat/etc/issue* (melihat versi OS dan kernel).
16. Menyalin file github yang telah didapatkan dengan menggunakan *command* awal wget.
17. Mencari *binaries* yang mendapatkan izin SUID.
18. Lalu, setelah mendapatkan *binaries* yang tepat, *import binaries* tersebut.
19. Selanjutnya, cek kembali Id didalam SSH untuk melihat *binaries* sudah tersimpan atau belum.
20. Setelah itu, masuk ke dalam mode *root* untuk mendapatkan *privilege environment access* dari VM Toppo ini.
21. Setelah berhasil melakukan akses ke dalam *root*, maka *privilege environment access* berhasil ditemukan.



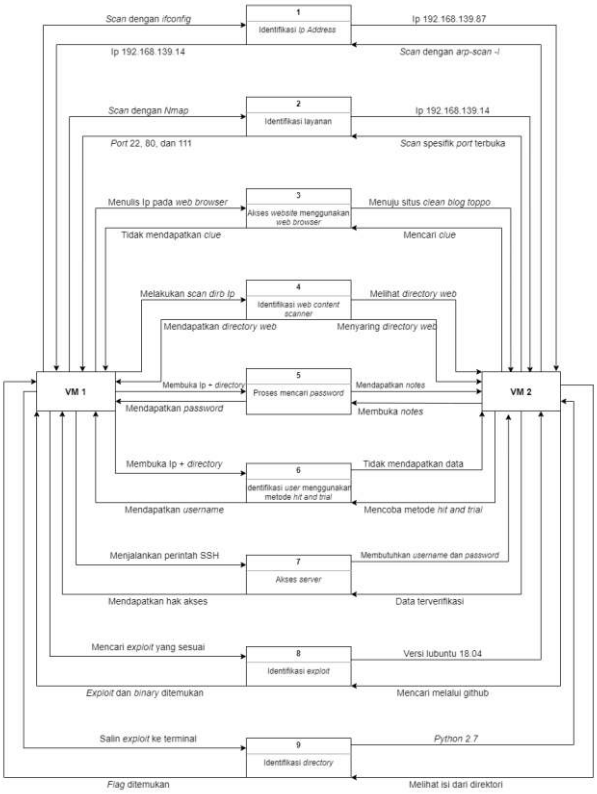
GAMBAR IV.3
ACTIVITY DIAGRAM

F. *Data Flow Diagram*

Data flow diagram merupakan tahapan-tahapan yang menggambarkan aliran data atau pertukaran data pada setiap proses yang berjalan. Pada penelitian ini akan di uji coba *walkthrough* yang berasal dari Hackersploit.

1. Tahap pertama lakukan *scan* IP *address* pada *Main* OS terlebih dahulu dengan *command ifconfig* dan didapatkan IP 192.168.139.87. *Command ifconfig* digunakan untuk mencari IP *address* pada sebuah OS.
2. Tahap ke dua mencari IP *address* dari *vulnerable machine* dengan menggunakan *command* arp-scan -l dan didapatkan IP Toppo: 1 192.168.139.14. *Command* arp-scan -l digunakan untuk melihat identitas dari sebuah *host* yang terhubung ke dalam suatu jaringan, misal *vulnerable machine* yang sedang terhubung ke OS.
3. Tahap ke tiga melakukan *scan* IP *address vulnerable machine* Toppo: 1 menggunakan *command* nmap -sC -sV -O -A 192.168.139.14 yang berguna untuk memperlihatkan koneksi *port* TCP yang terbuka, meningkatkan tingkat level pencarian IP, dan menemukan *port-port* yang terbuka yaitu, 22, 80, dan 111.
4. Tahap ke empat tuliskan IP *address vulnerable machine* Toppo: 1 ke dalam web *browser* yang terdapat di *operating system* Lubuntu dan pada web *browser* akan ditampilkan halaman utama dari Toppo: 1.
5. Tahap ke lima membuka *source code* pada halaman utama Toppo: 1 dan tidak mendapatkan *clue* dari *source code* halaman utama tersebut.
6. Tahap ke enam yaitu, pada halaman utama web Toppo tidak mendapatkan *clue* sama sekali, maka dilakukan pencarian *directory* dengan menggunakan *web content scanner* dengan *command* dirb http://192.168.139.14/. Dirb ini digunakan untuk melakukan pengecekan *directory* pada sebuah *website*.
7. Tahap ke tujuh akan ditampilkan beberapa *directory* dari IP *address* 192.168.139.14 dan lakukan penyaringan *directory* dari IP *address* tersebut agar mendapatkan *clue* untuk tahap selanjutnya.
8. Tahap ke delapan. Setelah dilakukan penyaringan *directory*, maka akan didapatkan *directory* yang sesuai yaitu http://192.168.139.14/admin/ dan salin *directory* tersebut, kemudian tempelkan ke dalam *browser*.
9. Tahap ke sembilan akan ditampilkan halaman web yang berisikan sebuah *notes* dengan format txt dan mencoba untuk membuka *notes* tersebut.
10. Tahap ke sepuluh buka *notes* tersebut dan didapatkan *clue* berupa *password* 12345ted123 untuk ke tahap selanjutnya.
11. Tahap sebelas menjalankan perintah SSH dengan asumsi *username* dari Toppo tersebut adalah ted dan SSH ini digunakan untuk mengakses jaringan yang terenkripsi.

12. Tahap dua belas *login* melalui SSH dengan *command* ssh ted@192.168.139.14 dengan *password* 12345ted123.
13. Tahap tiga belas mengecek terlebih dahulu apakah terdapat *exploit* di dalam Toppo SSH *interface* dengan menggunakan *command* id dan mengecek apakah sudah berada di dalam server Toppo atau belum dengan *command cat/etc/issue*.
14. Tahap empat belas mencari *binaries* yang tepat untuk *vulnerable machine* dengan cara wget dari salah satu github yaitu, mzet-/linux-exploit-suggester.
15. Tahap lima belas salin *command* yang didapatkan dari github dan lakukan *command wget https://raw.githubusercontent.com/mzet-/linux-exploit-suggester/master/linux-exploit-suggester.sh -O les.sh*. Kemudian jalankan *command find / -perm -u=s -type f 2>/dev/null*, digunakan untuk mencari *binaries* yang memiliki izin SUID. SUID (*Set Owner User ID*) adalah *special permissions* untuk dapat melakukan eksekusi file yang bisa dilakukan oleh pengguna lain dengan izin efektif dari pemilik file.
16. Tahap enam belas didapatkan *binaries python 2.7* dan lakukan *import binaries python* tersebut dengan menggunakan *command /usr/bin/python2.7 -c 'import pty;pty.spawn("/bin/sh")'* dan cek id tersebut.
17. Tahap tujuh belas masuk ke dalam mode *root* dengan menggunakan *command cd/root/* dan cek file tersebut dengan menggunakan *command ls.*
18. Tahap terakhir tuliskan *command cat flag.txt* dengan begitu *privileged environment access* didapatkan.



GAMBAR IV.4
DATA FLOW DIAGRAM

G. Pengukuran *Time Walkthrough*

Pengukuran *time* merupakan aktivitas mengamati, mengukur, dan mencatat waktu pada sebuah *walkthrough*. Waktu yang dicatat pada sub-bab ini adalah *real, user, system,* dan *host scanned time.* Berikut ini merupakan rincian tabel pengukuran *time* dari *walkthrough* 1 sebagai berikut:

TABEL IV.4
PENGUKURAN *TIME WALKTHROUGH*

Step (Command)	*Time Walkthrough 1 (s)*			*Hosts Scanned Time*
	Real	*User*	*Sys*	
Ifconfig	0,021	0,002	0	*null*
Arp-scan -l	1,93	0,138	0,012	1,906
Nmap	14,658	0,086	0,066	14,56
Web browser + IP *Address*	0	4	0	*null*
Source code web browser	0	1	0	*null*
Scan dirb	2m18,288	27,714	14,417	*null*
IP + *admin* + *notes*	0	2,26	0	*null*
Login SSH	6,752	0,538	0	*null*
Id	0,002	0	0	*null*
Cat /etc/issue	0,003	0	0	*null*
Wget github	1,812	0,06	0,032	*null*
Ls	0,002	0	0	*null*
Find binaries	4,096	0,104	0,468	*null*
Import binaries	0,032	0	0	*null*
Id	0,002	0	0	*null*
Mode root	0,003	0	0	*null*
Cat flag	0,267	0	0	*null*
Total	**167,868**	**35,816**	**14,995**	**16,466**

Null: Nilai tidak muncul pada hasil *host scanned time*

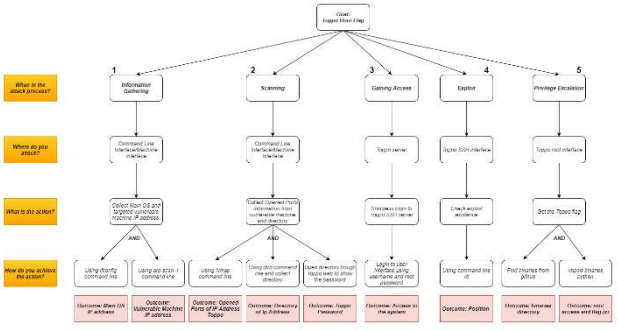
Bedasarkan Tabel IV.4 pengukuran time pada *walkthrough* 1 didapatkan total waktu dalam satuan detik sebagai berikut:

1. *Real time* : 167,868 s
2. *User time* : 35,816 s
3. *System time* : 14,995 s
4. *Host scanned time* : 16,466 s

V. ANALISIS

A. *Attack Tree*

Attack tree merupakan sebuah bentuk diagram konseptual untuk memahami suatu masalah pada proses serangan. *Attack tree* pada sub-bab ini dibuat berdasarkan jurnal konferensi dengan judul *CubeSat Security Attack Tree Analysis* (Falco, Santangelo & Viswanathan, 2021).



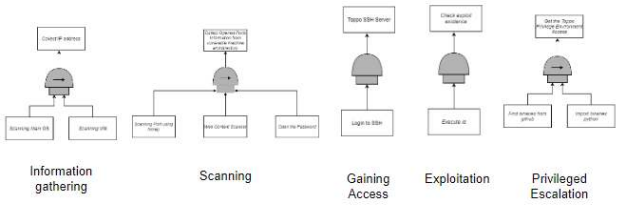
GAMBAR V.1
ATTACK TREE BERDASARKAN *WALKTHROUGH* 1

Secara umum *attack tree* dalam penelitian ini memiliki rincian penjelasan sebagai berikut:

1. *What is the goal?*
   Bertujuan untuk mengambil alih sistem dari *vulnerable machine* Toppo: 1 dengan mengakses *root* hanya sebatas sampai mendapatkan *privileged environment access.*
2. *Where do you attack*?
   Penyerangan ini dilakukan pada level sistem operasi pada *vulnerable machine* Toppo: 1.
3. *What is the action*?
   Mengambil alih sistem dan mendapatkan *privileged environment access*
4. *How do you achieve the action*?
   Aksi yang dilakukan untuk mendapatkan *privileged environment access* yaitu mencari IP *address main* OS dan *vulnerable machine*, memindai IP *address* dan menemukan *port* yang terbuka, mencari direktori menggunakan *web content scanner* dan mendapatkan *password*, masuk melalui SSH, memeriksa *exploit* pada *vulnerable machine*, dan mencari *binaries* serta melakukan impor *binaries* tersebut.
5. *Outcome*
   Outcome pada penelitian ini terbatas pada saat lingkungan *privileged environment access* pada *vulnerable machine* Toppo: 1 dapat diambil alih oleh penyerang.

Tahap penyusunan dan analisis *attack tree* berdasarkan *walkthrough* ini bertujuan untuk mengetahui tahapan-tahapan dan penggambaran berdasarkan *attack tree*.

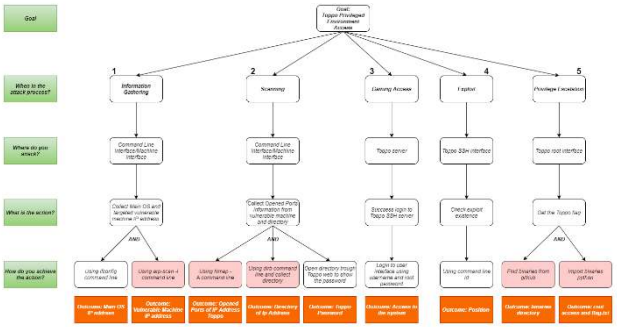
V.A.1 *Attack Tree* Pendekatan SAND *gate*



GAMBAR V.2
ATTACK TREE SAND *GATE*

1. *Information Gathering*
   Attack tree yang digambarkan pada *information gathering* menggunakan SAND *gate* untuk memberikan informasi bahwa urutan *attack tree* pada *information gathering*, dimulai dari kiri ke kanan pada tiap cabang. Pada *information gathering* ini hal yang harus dilakukan pertama kali yaitu harus *scanning Main* OS terlebih dahulu dan setelah itu *scanning* VM untuk mencapai *node* "*Collect IP address*" yang saling terhubung menggunakan SAND *gate*.
2. *Scanning*
   Attack tree yang digambarkan pada *scanning* menggunakan SAND *gate* untuk memberikan informasi bahwa urutan *attack tree* pada *scanning*, dimulai dari kiri ke kanan pada tiap cabang. Pada *attack tree scanning* memiliki 3 cabang yaitu, tahap pertama *scanning port using* nmap, tahap ke dua *web content scanner*, dan tahap terakhir *open the password*, untuk mencapai *node* "*Collect opened port information from vulnerable machine and directory*" yang saling terhubung menggunakan SAND *gate*.
3. *Gaining Access*
   Tahap *gaining access* bertujuan untuk masuk ke SSH. *Command line* yang digunakan untuk *login* melalui SSH adalah ssh ted@*username*.
4. *Exploit*
   Tahap *exploit* bertujuan untuk mengetahui apakah ketika *login* melalui SSH sudah terdapat *exploit* di dalamnya. *Command line* yang digunakan untuk mengetahui *exploit* adalah id.
5. *Privileged Escalation*
   Attack tree yang digambarkan pada *privileged escalation* menggunakan SAND *gate* untuk memberikan informasi bahwa urutan *attack tree* pada *privileged escalation,* dimulai dari kiri ke kanan pada tiap cabang. Pada *privileged escalation* ini hal yang harus dilakukan pertama kali yaitu harus *find binaries from* github terlebih dahulu dan setelah itu *import binaries python* untuk mencapai *node "Get Toppo Privileged Environment Access"* yang saling terhubung menggunakan SAND *gate.*

V.A.2 Penggambaran *Attack Tree* Berdasarkan 5 *Walkthrough*



GAMBAR V.3
ATTACK TREE BERDASARKAN 5 *WALKTHROUGH*

Berdasarkan Gambar V.3 dapat dijelaskan bahwa, dalam penggambaran *attack tree* berdasarkan 5 *walkthrough* akan dilakukan pengambilan *attack tools* dan *binaries* dengan waktu tercepat.

B. Analisis Perbandingan Berdasarkan *Metrics*

Pada sub-bab ini akan dijelaskan perbandingan antara lima buah *attack tree* berdasarkan *metrics* (*time, cost, and frequency*).

1. Analisis Perbandingan Berdasarkan *Time Metric*

Time metric merupakan sebuah pengukuran interval waktu yang dapat digunakan pada saat menjalankan *command line* berdasarkan *system metric. System metric* merupakan sebuah sistem satuan pengukuran desimal berstandar internasional.

Dikarenakan *user time* dan *system time* merupakan bagian dari *real time* maka, hanya *real time* saja yang dilakukan analisis. Perhitungan *time metric* menggunakan akumulasi dari *real time* yang digunakan pada *attack tree.* Berikut ini merupakan rumus untuk mengetahui nilai akhir dari *metric real time*:

$$\sum_{i=1}^{n} r = r_1 + r_2 + \ldots + r_n \quad \ldots\ldots\ldots\ldots(i)$$

dengan:
r = *real time*

n = batas teratas penjumlahan

Berikut ini merupakan Tabel V.B.1 berisi pemeringkatan *attack tree* berdasarkan *time metric:*

TABEL V.1
PEMERINGKATAN *ATTACK TREE* BERDASARKAN *TIME METRIC*

Peringkat	*Attack Tree*	*Time metric* (s)
1	*Attack Tree* 1	167,868
2	*Attack Tree* 5	180,126
3	*Attack Tree* 3	200,105
4	*Attack Tree* 4	262,528
5	*Attack Tree* 2	279,206

Berdasarkan Tabel V.1 dapat disimpulkan bahwa, waktu yang ditempuh untuk mendapatkan *privileged environment access* dari *vulnerable machine* Toppo: 1 berdasarkan *real time, attack tree* 1 merupakan peringkat pertama dengan waktu 167,868*s*. *Attack tree* 1 memiliki waktu yang lebih cepat dibandingkan dengan yang lain dikarenakan faktor *Attack Tools* yang digunakan, *attack tools* tersebut memiliki waktu yang lebih singkat untuk mendapatkan data. *Attack tree* 1 juga memiliki tahapan-tahapan yang relatif lebih sedikit jika dibandingkan dengan *walkthrough* yang lain.

2. Analisis Perbandingan Berdasarkan *Cost Metric*

Cost metric merupakan sebuah nilai yang dihitung atau digunakan pada suatu proses dari jumlah langkah *walkthrough*. Nilai dari *cost metric* ini didapatkan berdasarkan dari jumlah langkah pada setiap *walkthrough* dari *activity diagram.*

Berikut ini merupakan Tabel V.B.2 berisi pemeringkatan *attack tree* berdasarkan *cost metric:*

TABEL V.2
PEMERINGKATAN *ATTACK TREE* BERDASARKAN *COST METRIC*

Peringkat	*Attack Tree*	*Cost metric* (*step*)
1	*Attack Tree* 1	21
2	*Attack Tree* 5	22
3	*Attack Tree* 2	23
	Attack Tree 4	
4	*Attack Tree* 3	24

Berdasarkan Tabel V.2 dapat disimpulkan bahwa *attack tree* 1 merupakan peringkat pertama pada *attack tree* yang memiliki *cost* 21. *Attack tree* 1 ini memiliki jumlah tahapan-tahapan untuk menemukan *privileged environment access* relatif paling sedikit jika dibandingkan dengan *attack tree* yang lain. Tahapan-tahapan menemukan *privileged environment access* yang menyebabkan nilai dari *cost metric attack tree* 1 lebih sedikit jika dibandingkan dengan 4 *attack tree* lainnya.

3. Analisis Perbandingan Berdasarkan *Frequency Metric*

Frequency metric merupakan jumlah penggunaan *tools* yang digunakan dalam proses penyerangan. Pada sub-bab ini frekuensi berguna untuk melakukan perhitungan jumlah seberapa sering *tools* digunakan pada setiap *attack tree*.
Untuk mendapatkan nilai *f* maka, digunakan rumus frekuensi relatif sebagai berikut:

$$f = \frac{Jumlah\ Penggunaan\ Tools\ X}{Total\ Penggunaan\ Tools\ X\ Keseluruhan}\ X\ 100\% \ldots\ldots\ldots (ii)$$

dengan:
f = frekuensi relatif penggunaan *tools* dari *attack tree* yang sudah dijalankan (%)
Tools X= *tools* yang digunakan pada *attack tree*

Berikut ini merupakan Tabel V.3 yang berisi distribusi frekuensi dari *attack tree* sebagai berikut:

TABELV.3
DISTRIBUSI FREKUENSI *TOOLS* PADA *ATTACK TREE*

Attack Tree	*Tools*			
	Arp-scan	Netdiscover	Dirb	Nikto
Attack Tree 1	10%	*not used*	10%	*not used*
Attack Tree 2	*not used*	10%	10%	*not used*
Attack Tree 3	10%	*not used*	*not used*	10%
Attack Tree 4	*not used*	10%	10%	*not used*
Attack Tree 5	10%	*not used*	10%	*not used*
Total	**30%**	**20%**	**40%**	**10%**
Total Keseluruhan	**100%**			

Dari Tabel perhitungan *frequency tools* dapat dilakukan pemeringkatan sebagai berikut:

TABEL V.4
PEMERINGKATAN *TOOLS ATTACK TREE*

Peringkat	*Tools*	Frekuensi
1	Dirb	40%
2	Arp-scan	30%
3	Netdiscover	20%
4	Nikto	10%

Berdasarkan Tabel V.4 dapat disimpulkan bahwa, *attack tools* dirb dan arp-scan merupakan *attack tools* yang paling sering digunakan oleh penyerang. *Attack tools* dirb digunakan untuk mencari direktori IP *address* dengan nilai *frequency* 40%. *Attack tools* arp-scan digunakan untuk mencari IP *address vulnerable machine* dengan nilai *frequency* 30%. Jika dilakukan perbandingan maka, *attack tools* dirb lebih sering digunakan oleh penyerang dibandingkan dengan nikto. Sedangkan *attack tools* arp-scan lebih sering digunakan oleh penyerang dibandingkan dengan netdiscover. Dirb dan arp-scan lebih sering digunakan karena dalam pencarian informasi atau data ke dua *attack tools* memiliki waktu yang lebih singkat dibandingkan dengan nikto dan netdiscover.

V.B.4 Analisis Perbandingan Berdasarkan Penggabungan *Time Metric* dan *Cost Metric*

Berikut ini merupakan tabel yang berisikan pemeringkatan *attack tree* berdasarkan penggabungan hasil akhir *time metric* dan *cost metric*:

TABEL II.5
PEMERINGKATAN BERDASARKAN PENGGABUNGAN *TIME METRIC* DAN *COST METRIC*

Peringkat	*Attack Tree*	*Time metric* (s)	*Cost metric* (*step*)
1	*Attack Tree* 1	167,868	21
2	*Attack Tree* 5	180,126	22
3	*Attack Tree* 3	200,105	24

4	*Attack Tree* 4	262,528	23
5	*Attack Tree* 2	279,206	23

Berdasarkan Tabel V.5 dapat dijelaskan bahwa, pemeringkatan penggabungan *time metric* dan *cost metric* yang paling utama didahulukan adalah waktu tercepat. Dikarenakan dengan adanya waktu tercepat proses penyerangan akan memiliki waktu yang relatif singkat, dan pengembang VM target tidak mempunyai waktu yang cukup untuk menutup celah yang telah terbuka oleh penyerang. Sehingga probabilitas berhasilnya proses penyerangan ke mesin terget semakin tinggi.

VI. KESIMPULAN

Penerapan *walkthrough* dapat digunakan untuk menghasilkan tahapan-tahapan eksploitasi penyerangan pada *vulnerable machine* Toppo: 1 dan dapat digambarkan dengan *activity diagram* dan *data flow diagram*. Perumusan *attack tree* dapat dilakukan dengan menggunakan kerangka Cubesat *Security Attack Tree Analysis* dan SAND *gate* dari implementasi sebuah *walkthrough. Attack tree* 1 merupakan yang tercepat secara relatif jika dibandingkan dengan *attack tree* lain berdasarkan hasil perhitungan *time metric* dengan nilai 167,868 detik. *Attack tree* 1 memiliki tahapan tersingkat berdasarkan perhitungan *cost metric* dengan nilai 21. Dirb dan arp-scan merupakan *attack tools* yang memiliki persentase terbesar dalam penggunaannya berdasarkan perhitungan *frequency metric* dengan nilai 40% dan 30% secara berurutan.

REFERENSI

[1] M. Utomo, A. H. N. Ali and I. Affandi, "Pembuatan Tata Kelola Keamanan Informasi Kontrol Akses Berbasis ISO/IEC 27001:2005 Pada Kantor Pelayanan Perbendaharaan Surabaya I," *JURNAL TEKNIK ITS Vol. 1, No. 1, (Sept. 2012) ISSN: 2301-9271* , pp. 288-289, 2012.

[2] Diskominfo, "Hacker dengan Cracker," 18 October 2018. [Online]. Available: https://diskominfo.badungkab.go.id/artikel/18221-perbedaan-hacker-dengan-cracker.

[3] T. R. Ingoldsby, Attack Tree-based Threat Risk Analysis, Canada: Amenaza Technologies Limited, 2021.

[4] L. Kuipers, "Analysis of Attack Trees: fast algorithms for subclasses," *Bachelor thesis Computing Science,* pp. 9-19, 2020.

[5] A. Ramadhani, "Keamanan Informasi," *Journal of Information and Library Studies Vol. 1 No. 1,* pp. 40-41, 2018.

[6] AVG, "What Is an Exploit in Computer Security?," 22 October 2020. [Online]. Available: https://www.avg.com/en/signal/computer-security-exploits.

[7] P. P. Putra, "Pengembangan Sistem Keamanan Jaringan Menggunakan Rumusan Snort Rule (Hids) untuk Mendeteksi Serangan Nmap," *SATIN - Sains dan Teknologi Informasi, Vol. 2, No. 1, Juni 2016,* pp. 17-18, 2016.

[8] B. Light, J. Burgess and S. Duguay, "The walkthrough method: An approach to the study of apps," *new media & society 2018, Vol. 20(3) 881–900,* p. 882, 2018.

[9] P. Sulistyorini, "Pemodelan Visual dengan Menggunakan UML dan Rational Rose," *Jurnal Teknologi Informasi DINAMIK Volume XIV, No.1, Januari 2009 : 23-29,* pp. 24-27, 2009.

[10] Q. Li and Y. Chen, Modeling and Analysis of Enterprise and Information Systems: From Requirements to Realization 2009th, Springer, 2009, p. 85.

[11] Faradiba, Buku Materi Pembelajaran Metode Pengukuran Fisika, Jakarta Timur: PROGRAM STUDI PENDIDIKAN FISIKA FAKULTAS KEGURUAN DAN ILMU PENDIDIKAN UNIVERSITAS KRISTEN INDONESIA , 2020, 2020.

[12] I. M. A. Pradnyana, A. A. J. Permana and I. M. Putrama, "IMPLEMENTASI KONSEP PERANCANGAN MODEL KONSEPTUAL BASIS DATA STUDI KASUS: PERANCANGAN BASIS DATA SISTEM INFORMASI ADMINISTRASI BEASISWA DI UNDIKSHA," *Seminar Nasional Vokasi dan Teknologi (SEMNASVOKTEK),* p. 90, 2017.