



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



**ANALISIS SISTEM KEAMANAN PADA APACHE
WEB SERVER MENGGUNAKAN SELF SIGNED CA
DI WEBMIN**

SKRIPSI

FAUZAN NUGROHO

1807422024

**POLITEKNIK
NEGERI
JAKARTA**

**PROGRAM STUDI TEKNIK MULTIMEDIA DAN JARINGAN
JURUSAN TEKNIK INFORMATIKA DAN KOMPUTER
POLITEKNIK NEGERI JAKARTA**

2025



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



**ANALISIS SISTEM KEAMANAN PADA APACHE
WEB SERVER MENGGUNAKAN SELF SIGNED CA
DI WEBMIN**

SKRIPSI

**Dibuat untuk Melengkapi Syarat-Syarat yang Diperlukan
Untuk Memperoleh Diploma Empat Politeknik**

FAUZAN NUGROHO

1807422024

**PROGRAM STUDI TEKNIK MULTIMEDIA DAN JARINGAN
JURUSAN TEKNIK INFORMATIKA DAN KOMPUTER
POLITEKNIK NEGERI JAKARTA**

2025

Jurusan Teknik Informatika dan Komputer – Politeknik Negeri Jakarta



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

SURAT PERNYATAAN BEBAS PLAGIARISME

Saya yang bertanda tangan di bawah ini:

Nama : Fauzan Nugroho

NIM : 1807422024

Jurusan/Program Studi : T.Informatika dan Komputer / Teknik Multimedia dan

Jaringan

Judul skripsi : Analisis Sistem Keamanan pada Apache Web Server

Menggunakan Self-signed CA di Webmin

Menyatakan dengan sebenarnya bahwa skripsi ini benar-benar merupakan hasil karya saya sendiri, bebas dari peniruan terhadap karya dari orang lain. Kutipan pendapat dan tulisan orang lain ditunjuk sesuai dengan cara-cara penulisan karya ilmiah yang berlaku.

Apabila di kemudian hari terbukti atau dapat dibuktikan bahwa dalam skripsi ini terkandung ciri-ciri plagiat dan bentuk-bentuk peniruan lain yang dianggap melanggar peraturan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Depok, 6 Juli 2025

Yang membuat pernyataan



Fauzan Nugroho

NIM 1807422024



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

LEMBAR PENGESAHAN

Skripsi diajukan oleh :

Nama : Fauzan Nugroho

NIM : 1807422024

Program Studi : Teknik Multimedia dan Jaringan

Judul Skripsi : Analisis Sistem Keamanan pada Apache Web Server

Menggunakan Self-signed CA di Webmin

Telah diuji oleh tim penguji dalam Sidang Skripsi pada hari Selasa,

Tanggal 8, Bulan Juli, Tahun 2025 dan

Dinyatakan **LULUS**.

Disahkan oleh

Pembimbing I

: Defiana amaldy, S. Tp., M.Si.

Penguji I

: Dr. Prihatin Oktivasari, S.Si., M.Si

Penguji II

: Fachroni Arbi Murad, S.Kom., M.Kom

Penguji III

: Ariawan Andi Suhandana, S.Kom., M.Ti

**POLITEKNIK
NEGERI
JAKARTA**

Mengetahui :

Jurusan Teknik Informatika dan Komputer

Ketua



Dr., Anita Hidayati, S.Kom., M.Kom.

NIP. 197908032003122003



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

KATA PENGANTAR

Puji syukur kepada Allah SWT berkat Rahmat, Hidayah, dan Karunia-Nya kepada kita semua sehingga penulis dapat menyelesaikan skripsi dengan judul ‘Analisis Sistem Keamanan pada Apache Web Server Menggunakan Self-signed CA di Webmin’ . Penulis menyadari dalam penyusunan skripsi ini tidak akan selesai tanpa bantuan dari berbagai pihak. Karena itu pada kesempatan ini penulis ingin mengucapkan terima kasih kepada:

1. Allah SWT tuhan yang maha esa, yang telah memberikan saya rizki berupa kesehatan dan akal pikiran yang sangat berharga bagi penulis sehingga laporan ini dapat terselesaikan dengan baik.
2. Orang tua dan keluarga penulis yang telah memberikan bantuan dukungan secara moral dan material.
3. Bapak Defiana arnaldy, S. Tp., M. Si. selaku pembimbing skripsi yang telah membimbing penulis dalam menyelesaikan skripsi ini.

Penulis menyadari skripsi ini tidak luput dari berbagai kekurangan. Penulis mengharapkan saran dan kritik demi kesempurnaan dan perbaikannya sehingga akhirnya laporan skripsi ini dapat memberikan manfaat bagi bidang pendidikan dan penerapan di lapangan serta bisa dikembangkan lagi lebih lanjut.

Depok, 6 Juli 2025


(Fauzan Nugroho)



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

SURAT PERNYATAAN PERSETUJUAN PUBLIKASI SKRIPSI UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademik Politeknik Negeri Jakarta, saya bertanda tangan dibawah ini :

Nama : Fauzan Nugroho

NIM : 1807422024

Jurusan/Program Studi : T.Informatika dan Komputer / Teknik Multimedia dan Jaringan

Demi pengembangan ilmu pengetahuan , menyetujui untuk memberikan kepada Politeknik Negeri Jakarta Hak Bebas Royalti Non-Eksklusif atas karya ilmiah saya yang berjudul :

Analisis Sistem Keamanan pada Apache Web Server Menggunakan Self-signed CA di Webmin.

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non-Eksklusif ini Politeknik Negeri Jakarta Berhak menyimpan, mengalihmediakan/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan skripsi saya tanpa meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta..

Demikian pernyataan ini saya buat dengan sebenarnya.

Depok, 6 Juli 2025

Yang Menyatakan



1807422024



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Analisis Sistem Keamanan pada Apache Web Server Menggunakan Self-signed CA di Webmin

Abstrak

Komunikasi data melalui protokol HTTP rentan terhadap serangan penyadapan, sementara penggunaan sertifikat dari Otoritas Sertifikat (CA) publik seringkali tidak praktis untuk lingkungan pengembangan lokal. Penelitian ini menganalisis efektivitas implementasi sertifikat yang ditandatangani sendiri (self-signed certificate) pada Apache Web Server yang dikelola melalui Webmin sebagai solusi keamanan. Metode penelitian yang digunakan adalah eksperimen dalam lingkungan virtual terisolasi yang mencakup server (Ubuntu), klien (Windows), dan penyerang (Kali Linux). Ketahanan sistem diuji melalui simulasi serangan Man-in-the-Middle (MITM) untuk membandingkan keamanan koneksi sebelum (HTTP) dan sesudah (HTTPS) implementasi sertifikat. Hasil pengujian menunjukkan bahwa koneksi HTTPS yang diamankan dengan sertifikat self-signed berhasil mengenkripsi seluruh lalu lintas data. Berbeda dengan koneksi HTTP yang memungkinkan data disadap dalam bentuk plaintext, koneksi terenkripsi ini memastikan data hanya terbaca sebagai ciphertext oleh penyerang, sehingga kerahasiaan dan integritasnya terjaga. Disimpulkan bahwa penggunaan sertifikat self-signed merupakan solusi keamanan untuk melindungi lingkungan pengembangan atau pengujian internal dari ancaman penyadapan data menggunakan metode MITM, meskipun tidak direkomendasikan untuk server produksi yang diakses publik.

Kata Kunci: http, https, Self-signed CA, website.



DAFTAR ISI

SURAT PERNYATAAN BEBAS PLAGIARISME	ii
LEMBAR PENGESAHAN	iii
KATA PENGANTAR.....	iv
SURAT PERNYATAAN PERSETUJUAN PUBLIKASI	v
SKRIPSI UNTUK KEPENTINGAN AKADEMIS	v
Abstrak.....	vi
DAFTAR ISI.....	vii
DAFTAR GAMBAR.....	ix
DAFTAR TABEL	xi
BAB I.....	1
PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah.....	3
1.3 Batasan Masalah.....	3
1.4 Tujuan dan Manfaat.....	4
1.4.1 Tujuan	4
1.4.2 Manfaat	4
1.5 Sistematika Penulisan.....	5
BAB II	6
TINJAUAN PUSTAKA	6
2.1 Penelitian Sejenis	6
2.2 Tinjauan Pustaka	7
2.3 Sistem Keamanan	9
2.4 Keamanan Website.....	10
2.5 Ubuntu Server.....	10
2.6 Web Server	11
2.7 Apache Web Server	12
2.8 Webmin	13
2.9 Self-signed CA	14
2.10 Perbedaan HTTP dengan HTTPS.....	15
2.11 Man-In-the-Middle	16
BAB III.....	18
METODE PENELITIAN	18
3.1 Rancangan Penelitian	18
3.2 Tahapan Penelitian	18
3.2.1 Latar Belakang	21
3.2.2 Observasi.....	21
3.2.3 Perancangan	21
3.2.4 Pengujian.....	21

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

3.3	Objek Penelitian	23
BAB IV		24
HASIL DAN PEMBAHASAN		24
4.1	Analisis Kebutuhan	24
4.1.1	Kebutuhan Perangkat Keras	24
4.1.2	Kebutuhan Perangkat Lunak	25
4.1.3	Kebutuhan Konfigurasi	27
4.2	Perancangan Sistem Keamanan	27
4.3	Implementasi Self-signed CA pada Apache via Webmin	32
4.3.1	Persiapan Lingkungan Server dan Prasyarat	34
4.3.2	Persiapan Lingkungan Client	43
4.3.3	Instalasi dan Akses Webmin	50
4.3.4	Konfigurasi Prasyarat Web (DNS, Database, Apache Virtual Host) ..	52
4.3.5	Implementasi SSL/TLS Menggunakan Sertifikat Self-Signed via OpenSSL dan Webmin	56
4.4	Pengujian Sistem Keamanan	61
4.4.1	Deskripsi Pengujian	61
4.4.2	Prosedur Pengujian	62
4.4.3	Data Hasil Pengujian	67
4.4.4	Analisis Data dan Evaluasi Pengujian	71
BAB V		75
PENUTUP		75
5.1	Simpulan	75
5.2	Saran	77

**POLITEKNIK
NEGERI
JAKARTA**



DAFTAR GAMBAR

Gambar 2.1 <i>Ubuntu</i>	11
Gambar 2.2 <i>Web Server</i>	12
Gambar 2.3 <i>Apache Web Server</i>	13
Gambar 2.4 <i>Webmin</i>	14
Gambar 2.5 <i>HTTP dan HTTPS</i>	16
Gambar 3.1 <i>Flowchart Tahapan Penelitian</i>	19
Gambar 3.2 <i>Flowchart Self-Signed</i>	20
Gambar 3.3 <i>Flowchart Cara Kerja Man-in-the-middle</i>	22
Gambar 4.1 <i>Flowchart Perancangan Sistem Keamanan Web Server</i>	28
Gambar 4.2 <i>File Iso Ubuntu Server</i>	35
Gambar 4.3 <i>Ubuntu Virtual Machine Name</i>	35
Gambar 4.4 <i>Ubuntu Hardware Configuration</i>	35
Gambar 4.5 <i>Ubuntu Hard disk configuration</i>	36
Gambar 4.6 <i>Ubuntu Adapter Configuration</i>	36
Gambar 4.7 <i>Ubuntu Summary</i>	37
Gambar 4.8 <i>Grub Ubuntu</i>	37
Gambar 4.9 <i>Language Setting</i>	38
Gambar 4.10 <i>Keyboard Layout Ubuntu</i>	38
Gambar 4.11 <i>Base Installation Ubuntu</i>	39
Gambar 4.12 <i>Network Configuration Ubuntu</i>	39
Gambar 4.13 <i>Storage Ubuntu</i>	40
Gambar 4.14 <i>File System Ubuntu</i>	40
Gambar 4.15 <i>Username Setting</i>	40
Gambar 4.16 <i>OpenSSH Ubuntu</i>	41
Gambar 4.17 <i>Instalasi Akhir Ubuntu</i>	41
Gambar 4.18 <i>Update sistem ubuntu</i>	42
Gambar 4.19 <i>Upgrade sistem ubuntu</i>	42
Gambar 4.20 <i>Instalasi wget dan apttransporthttps</i>	42
Gambar 4.21 <i>Penambahan Repository Webmin</i>	43
Gambar 4.22 <i>Impor GPG Key</i>	43
Gambar 4.23 <i>Menambahkan aturan Firewall</i>	43
Gambar 4.24 <i>Mereload aturan Firewall</i>	43
Gambar 4.25 <i>File ISO Windows10</i>	44
Gambar 4.26 <i>Virtual Machine name Windows10</i>	44
Gambar 4.27 <i>Hardware Configuration Windows10</i>	45
Gambar 4.27 <i>Hard disk Configuration Windows10</i>	45
Gambar 4.28 <i>Pengaturan Adaptor Jaringan Internal Virtualbox</i>	46
Gambar 4.29 <i>Setup Windows10</i>	47
Gambar 4.30 <i>Install now Windows10</i>	47
Gambar 4.31 <i>Activation Windows10</i>	48
Gambar 4.31 <i>Custom Install Windows10</i>	48
Gambar 4.32 <i>Unallocated Space Windows10</i>	49
Gambar 4.33 <i>Installing Windows10</i>	49
Gambar 4.34 <i>Home Windows10</i>	50
Gambar 4.35 <i>Proses Instalasi Webmin</i>	50

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Gambar 4.36 Status Layanan Webmin Setelah Instalasi	51
Gambar 4.37 Tampilan Login Webmin	51
Gambar 4.38 Tampilan Dashboard Webmin	51
Gambar 4.39 Konfigurasi Master Zone	52
Gambar 4.40 Konfigurasi Name Server	52
Gambar 4.41 Pembuatan Database	53
Gambar 4.42 Mengatur Permission Database	53
Gambar 4.43 Membuat User	54
Gambar 4.44 Struktur Direktori Public HTML via File Manager Webmin	54
Gambar 4.45 Pembuatan Virtual Server	55
Gambar 4.46 Edit Virtual Host	55
Gambar 4.47 Konfigurasi Network dan Address	56
Gambar 4.48 Konfigurasi Webmin	57
Gambar 4.49 Certificate Authority	57
Gambar 4.50 Create New CA	57
Gambar 4.51 SSL Encryption	58
Gambar 4.52 Current Certificate	58
Gambar 4.53 New CA	59
Gambar 4.54 Certificate Detail	59
Gambar 4.55 Manage Certificate	59
Gambar 4.56 Import Certificates CA	60
Gambar 4.57 Browse Certificate CA	60
Gambar 4.58 Verifikasi awal SSL/TLS	61
Gambar 4.59 Indikator Koneksi HTTP yang tidak aman	62
Gambar 4.60 Indikator Koneksi HTTPS yang sudah aman	63
Gambar 4.61 File ISO Installer Kali Linux	64
Gambar 4.62 Bios mode Kali Linux	64
Gambar 4.63 Installer Kali GNU/Linux	65
Gambar 4.64 Konfigurasi Keyboard	65
Gambar 4.65 Pembuatan Password Kali Linux	66
Gambar 4.66 Pemilihan Partisi Instalasi Kali Linux	66
Gambar 4.67 Pemilihan partisi instalasi GRUB	67
Gambar 4.68 Website HTTP	68
Gambar 4.69 Website HTTPS	68
Gambar 4.70 Hasil Ping dari Mesin Attacker ke Client dan Server	69
Gambar 4.71 Hasil Ping dari Mesin Server ke Attacker dan Client	69
Gambar 4.72 Hasil Penyadapan Data HTTP oleh Ettercap	70
Gambar 4.73 Tampilan Paket Terenkripsi TLSv1.3 di Wireshark	71
Gambar 4.74 Analisis Host di NetworkMiner dengan Lalu Lintas Terenkripsi	71



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

DAFTAR TABEL

Tabel 2.1. Perbandingan Tinjauan Peneliti	6
Tabel 4.1 Infrastruktur Mesin Virtual	33





Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

BAB I PENDAHULUAN

1.1 Latar Belakang

Teknologi jaringan komputer berkembang dengan sangat pesat, perkembangan ini diawali oleh munculnya internet sebagai media informasi (terkoneksi dari seluruh belahan dunia). Pengaruh dari kehadiran teknologi ini yaitu mempermudah manusia untuk mengakses tugas ataupun informasi yang diinginkan dan dengan mudahnya menyelesaikan pekerjaan secara instan (Momin & Ali, 2023). Namun dengan perkembangan teknologi tersebut seperti komputer ke internet dapat membuka potensi adanya lubang keamanan (*security hole*). Hal ini menjadi masalah besar apabila tidak ditangani dengan benar karena dengan tereksplotasinya lubang keamanan, seperti *hacker* dan *intruder* (penyusup) dapat dengan mudah melakukan tindak kejahatan dalam dunia cyber yang dikenal dengan *cyber crime*.

Website adalah salah satu aplikasi yang berisikan protokol HTTP (*hyper transfer protocol*) dan untuk mengakses perangkat lunak yang disebut browser. Fungsi website diantaranya : media promosi, pemasaran, informasi, pendidikan dan komunikasi (Nurlailah & Nova Wardani, 2023). Pada dasarnya terdapat 4 elemen dasar dari sebuah website yaitu : browser, server, URL dan *pages*. Beberapa tahun terakhir keamanan merupakan pokok persoalan utama. Banyak diberitakan mengenai penyusupan di website dan perusahaan atau perusahaan serta penghilangan aset perusahaan yang dalam bentuk digital (Mundt & Baier, 2023).

Web server sering kali menjadi target dari berbagai jenis serangan baik yang sifatnya minor maupun major sehingga berakibat fatal. Akan tetapi, website dijadikan pintu oleh peretas (*hacker*) untuk menembus *web server* bahkan sampai ke root untuk kesenangan bahkan dengan sengaja menyerang website tersebut.

Keamanan *server website* biasanya merupakan masalah dari seorang administrator. Seorang administrator bertugas untuk mengawasi dan melakukan tindakan preventif ketika terjadi aksi penyusupan dan serangan. Untuk itu keamanan dari sistem informasi yang digunakan harus terjamin dalam batas yang dapat diterima.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Terdapat berbagai cara untuk mengamankan *web server* dari serangan *hacker* salah satunya yaitu menggunakan SSL/TLS. *Secure Socket Layer* (SSL) adalah sebuah protokol keamanan digital yang memungkinkan proses komunikasi terenkripsi antara pengguna situs *web* dan *web browser*. Sementara itu, *Transport Layer Security* (TLS) adalah sebuah sistem keamanan yang berfokus pada perlindungan privasi dan integritas data selama transmisi informasi di jaringan. Dalam mengelola dan mengkonfigurasi *web server Apache*, khususnya terkait implementasi SSL/TLS, kompleksitas perintah baris seringkali menjadi tantangan bagi administrator.

Webmin hadir sebagai solusi berbasis *web* yang menyediakan antarmuka grafis intuitif untuk manajemen sistem UNIX, termasuk konfigurasi *web server* seperti *Apache*. Penggunaan Webmin secara signifikan mempermudah proses administrasi dan konfigurasi SSL/TLS, memungkinkan pengelolaan sertifikat, pengaturan virtual host, dan restart layanan dengan lebih efisien, bahkan bagi mereka yang kurang akrab dengan *command line*.

Namun, lingkungan pengujian lokal menghadapi kendala untuk memperoleh sertifikat dari Otoritas Sertifikat (CA) public karena memerlukan proses validasi domain yang harus dapat diakses secara publik. Sebagai solusinya, penelitian ini menggunakan sertifikat self-signed, yakni sertifikat yang ditandatangani oleh pembuatnya sendiri. Metode ini memungkinkan simulasi penuh lingkungan HTTPS terenkripsi di dalam jaringan lokal yang terisolasi untuk tujuan pengujian internal. Dengan demikian, penerapan self-signed CA menjadi langkah fundamental dalam penelitian ini untuk menganalisis cara kerja dan efektivitas SSL/TLS tanpa bergantung pada infrastruktur eksternal.

Berdasarkan permasalahan tersebut, maka penelitian yang akan dilakukan akan mengenai “Analisis Sistem Keamanan pada *Apache Web Server* Menggunakan *Self-Signed CA* di *Webmin*”. Sebuah website yang sudah menggunakan SSL atau TLS, meskipun self-signed, akan menunjukkan notifikasi tertentu di browser bahwa komunikasi sudah terenkripsi. SSL atau TLS juga secara umum mempengaruhi SEO dan peringkat website di mesin pencarian karena sesuai dengan algoritma yang ditetapkan Google, di mana Google cenderung melarang penggunaannya untuk mengunjungi website yang tidak memiliki atau menggunakan SSL atau TLS.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

1.2 Perumusan Masalah

Berdasarkan latar belakang, perumusan masalah penelitian yang akan dikaji adalah sebagai berikut:

- a. Bagaimana cara kerja Self-signed CA dalam mengamankan web server
- b. Bagaimana analisis keamanan dari Self-signed CA pada Webmin
- c. Seberapa efektif implementasi SSL/TLS dengan sertifikat self-signed pada Apache Web Server melalui Webmin dalam memitigasi serangan Man-in-the-Middle (MITM) di lingkungan pengujian?
- d. Apa saja kelebihan dan keterbatasan teknis dari penggunaan sertifikat self-signed CA sebagai solusi enkripsi lapisan transportasi data untuk Apache Web Server dalam konteks lingkungan pengujian lokal?

1.3 Batasan Masalah

Batasan masalah yang ada dalam penelitian ini adalah :

1. Penelitian ini fokus pada implementasi dan analisis keamanan SSL/TLS menggunakan sertifikat self-signed CA. Penggunaan sertifikat dari Certificate Authority (CA) publik atau pihak ketiga tidak termasuk dalam ruang lingkup penelitian ini, karena tujuan utamanya adalah menguji lingkungan lokal.
2. Web server yang digunakan dalam penelitian ini adalah Apache Web Server.
3. Manajemen dan konfigurasi web server, khususnya terkait SSL/TLS, akan dilakukan melalui Webmin. Metode konfigurasi manual melalui command line atau tool manajemen lain di luar Webmin tidak menjadi fokus pembahasan.
4. Lingkungan pengujian SSL/TLS dan simulasi serangan Man-in-the-Middle (MITM) dilakukan dalam jaringan virtual yang terisolasi menggunakan VirtualBox, dengan mesin Ubuntu sebagai server, Windows sebagai client, dan Kali Linux sebagai attacker. Penelitian ini tidak melibatkan pengujian di lingkungan produksi atau jaringan publik.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

5. Analisis keamanan difokuskan pada lapisan transportasi data (SSL/TLS) dan dampaknya terhadap kerahasiaan komunikasi serta ketahanan terhadap serangan MITM. Penelitian ini tidak membahas kerentanan aplikasi web secara spesifik seperti SQL Injection, XSS, dll. atau keamanan sistem operasi secara menyeluruh di luar konteks konfigurasi web server dan SSL/TLS.

1.4 Tujuan dan Manfaat

1.4.1 Tujuan

Tujuan pembuatan skripsi ini adalah untuk:

1. Mengimplementasikan sertifikat self-signed CA pada Apache Web Server melalui Webmin di lingkungan lokal untuk mengaktifkan komunikasi HTTPS terenkripsi.
2. Menganalisis perbandingan tingkat keamanan komunikasi data pada Apache Web Server antara kondisi HTTP dan HTTPS setelah implementasi sertifikat self-signed CA, dengan mengamati indikator keamanan browser dan lalu lintas jaringan.
3. Mengevaluasi efektivitas implementasi SSL/TLS dengan sertifikat self-signed CA dalam memitigasi serangan Man-in-the-Middle (MITM) pada Apache Web Server di lingkungan pengujian yang terisolasi.
4. Menganalisis kelebihan dan keterbatasan teknis dari penggunaan sertifikat self-signed CA sebagai solusi enkripsi lapisan transportasi data untuk Apache Web Server dalam konteks lingkungan pengujian lokal.

1.4.2 Manfaat

Manfaat dari skripsi ini dapat diuraikan menjadi dua aspek utama, yaitu manfaat secara akademis dan manfaat praktis.

Secara akademis, penelitian ini memberikan pengaruh terhadap pengembangan pengetahuan di dalam bidang keamanan *web server*, khususnya terkait konfigurasi dan penerapan sertifikat enkripsi.

Secara praktis, penelitian ini bermanfaat bagi berbagai jenis pengguna, baik *end-user* maupun pemilik atau pengelola server, dalam memahami serta meningkatkan



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

tingkat keamanan saat mengakses atau mengelola sistem berbasis web. Indikator rasa aman yang dimaksud meliputi perlindungan terhadap akses tidak sah, integritas data, dan keandalan sistem yang digunakan.

1.5 Sistematika Penulisan

Sistematika penulisan berikut dibentuk untuk mempermudah dalam penyusunan skripsi penelitian ini dengan penulisan yang baik ketika sistematika penulisan yang digunakan dijabarkan sebagai berikut:

BAB I PENDAHULUAN

Bab pendahuluan mendeskripsikan mengenai latar belakang masalah, rumusan masalah, batasan masalah, tujuan dan manfaat, serta sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Definisi dan penjelasan pustaka yang dijadikan referensi dalam penelitian ini akan dijelaskan pada bab 2 titik teori yang dipaparkan diantaranya mengenai keamanan website kemudian aplikasi yang digunakan untuk memakai website serta terus yang digunakan untuk enkripsi.

BAB III METODE PENELITIAN

Bab metode penelitian memaparkan pendekatan-pendekatan yang dapat dilakukan dalam perancangan dan implementasi tujuan penelitian secara terstruktur.

BAB IV HASIL DAN PEMBAHASAN

Bab ini berisi substansi meliputi analisis kebutuhan, perancangan, implementasi, pengujian serta hasil analisis pengujian.

BAB V PENUTUP

Bab ini berisi simpulan dan saran.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Sejenis

Tabel 2.1. Perbandingan Tinjauan Peneliti

Jurnal	Perbandingan Tinjauan Penelitian
SISTEM PENGAMANAN WEB SERVER DENGAN WEB APPLICATION FIREWALL - (Liu & Cheng, 2024) - Object : Web Application Firewall	Hasil penelitian menunjukkan modsecurity dapat mencegah serangan SQL injection dan xss dan audit konsol dapat digunakan sebagai mekanisme pelaporan serangan, dan mengetahui tingkat bahaya dari serangan tersebut
IMPLEMENTASI SISTEM KEAMANAN WEB SERVER MENGGUNAKAN PFSENSE (2020) - (Arman & Rachmat, 2020) - Object : PFSENSE	Sistem snort mampu mengetahui sebagai alat dan menyimpannya di dalam log seperti serangan ping of dead dan slowloris. Berdasarkan lock snort pfsense mampu melakukan pemblokiran otomatis dalam durasi tertentu.

Tabel 2.1 menyajikan perbandingan antara penelitian ini dengan beberapa penelitian sejenis yang relevan, guna memetakan posisi dan kebaruan dari pendekatan yang diambil dalam lingkup keamanan web server.

Penelitian pertama yang diangkat adalah karya Liu & Cheng (2024), yang berfokus pada pengamanan *web server* menggunakan *Web Application Firewall* (WAF). Hasil studi mereka menunjukkan bahwa implementasi *modsecurity* secara efektif mampu mencegah serangan pada lapisan aplikasi, khususnya *SQL injection* dan *Cross-Site Scripting* (XSS). Lebih lanjut, sistem tersebut didukung oleh konsol audit yang berfungsi sebagai mekanisme untuk melaporkan serangan sekaligus mengukur tingkat bahaya dari ancaman yang terdeteksi.

Penelitian kedua yang menjadi acuan adalah dari Arman & Rachmat (2020), yang mengimplementasikan sistem keamanan jaringan menggunakan pfSense. Temuan

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

utama mereka menunjukkan bahwa sistem pendeteksi intrusi (IDS) Snort yang terintegrasi pada pfSense mampu mendeteksi serangan seperti Ping of Death dan Slowloris, lalu menyimpannya di dalam log. Berdasarkan data log tersebut, pfSense mampu melakukan tindakan responsif berupa pemblokiran otomatis terhadap alamat IP penyerang untuk durasi tertentu.

Kedua penelitian tersebut menawarkan solusi keamanan yang berfokus pada lapisan aplikasi (WAF) dan lapisan jaringan (IDS/IPS). Hal ini berbeda dengan fokus penelitian saat ini, yang menganalisis keamanan pada lapisan transportasi data melalui implementasi enkripsi SSL/TLS menggunakan Self-signed CA yang dikelola via Webmin untuk melindungi komunikasi dari serangan penyadapan.

2.2 Tinjauan Pustaka

Perkembangan pesat teknologi jaringan dan internet telah mempermudah berbagai aspek kehidupan manusia, mulai dari akses informasi hingga penyelesaian pekerjaan secara instan (Momin & Ali, 2023). Namun, kemudahan ini diiringi dengan munculnya potensi celah keamanan (security hole) yang dapat dieksploitasi untuk tindak kejahatan siber atau cyber crime (Nugroho et al., 2024). Seiring meningkatnya ketergantungan pada teknologi digital, risiko kejahatan seperti peretasan, pencurian data, dan penyalahgunaan informasi juga terus meningkat, menjadikan pembangunan sistem keamanan yang kokoh sebuah keharusan (Tripathy, 2024).

Dalam ekosistem digital, web server menjadi komponen krusial yang sering kali menjadi target utama berbagai jenis serangan siber (Mundt & Baier, 2023). Perangkat lunak populer seperti Apache HTTP Server, meskipun fleksibel dan dapat diandalkan, tidak luput dari ancaman ini, sehingga memastikan keamanannya adalah tugas esensial bagi administrator sistem (Koch & Hao, 2021). Mengingat keamanan informasi pada sebuah situs web telah menjadi isu yang amat krusial, mengabaikannya dapat berdampak sangat merugikan, di mana peretas dapat dengan mudah mengambil data-data penting atau bahkan merusak tampilan web (Alby et al., 2022).

Berbagai penelitian telah mengeksplorasi metode untuk mengamankan web server. Salah satu pendekatan adalah menggunakan Web Application Firewall (WAF).



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Sebuah studi oleh Liu & Cheng (2024) menunjukkan bahwa implementasi WAF berbasis OpenResty dengan ModSecurity secara efektif dapat mencegah serangan umum seperti SQL injection dan Cross-Site Scripting (XSS). Sistem ini juga dilengkapi dengan konsol audit yang berfungsi sebagai mekanisme pelaporan untuk mengetahui tingkat bahaya dari serangan yang terjadi (Liu & Cheng, 2024).

Penelitian lain yang dilakukan oleh Arman & Rachmat (2020) berfokus pada implementasi sistem keamanan menggunakan pfSense yang terintegrasi dengan Snort sebagai Intrusion Detection System (IDS). Hasilnya menunjukkan bahwa sistem tersebut mampu mendeteksi dan mencatat berbagai jenis serangan, seperti Ping of Death dan Slowloris, ke dalam log. Berdasarkan catatan log tersebut, pfSense dapat secara otomatis melakukan pemblokiran terhadap alamat IP penyerang untuk durasi waktu tertentu, memberikan lapisan pertahanan yang proaktif (Arman & Rachmat, 2020).

Salah satu pilar utama dalam keamanan komunikasi web modern adalah penggunaan enkripsi melalui protokol Secure Socket Layer (SSL) atau penggantinya, Transport Layer Security (TLS). SSL/TLS adalah protokol keamanan digital yang memungkinkan proses komunikasi terenkripsi antara peramban web (client) dan web server (Aayush et al., 2022). Ketika sebuah situs web menggunakan SSL/TLS, protokol komunikasinya berubah dari HTTP (Hypertext Transfer Protocol) menjadi HTTPS, di mana 'S' berarti secure, menandakan bahwa seluruh data yang ditransmisikan dilindungi dari penyadapan (Fattah, 2021).

Perbedaan mendasar antara HTTP dan HTTPS terletak pada lapisan keamanan. HTTP mengirimkan data dalam bentuk plaintext yang rentan disadap, sedangkan HTTPS menggunakan enkripsi SSL/TLS untuk mengubah data menjadi ciphertext (Hu et al., 2021). Ini memastikan bahwa meskipun data berhasil dicuri oleh pihak ketiga, data tersebut tidak dapat dibaca atau dimengerti, sehingga menjaga kerahasiaan (confidentiality) dan keutuhan (integrity) informasi selama transmisi (Khan et al., 2022).

Untuk mengaktifkan HTTPS, sebuah web server memerlukan sertifikat digital. Dalam lingkungan produksi, sertifikat ini biasanya diperoleh dari Otoritas Sertifikat

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

(CA) publik yang terpercaya. Namun, untuk lingkungan pengembangan atau pengujian lokal, memperoleh sertifikat dari CA publik seringkali tidak praktis. Sebagai solusinya, digunakanlah sertifikat yang ditandatangani sendiri (self-signed certificate) (Nugroho et al., 2024). Sertifikat self-signed adalah sertifikat yang ditandatangani oleh pembuatnya sendiri, bukan oleh CA publik yang diakui secara global (Khan et al., 2022).

Meskipun peramban web akan menampilkan peringatan keamanan karena sertifikat ini tidak berasal dari sumber terpercaya, penggunaannya sangat relevan untuk simulasi penuh lingkungan HTTPS di jaringan lokal yang terisolasi (Mai et al., 2022). Fungsi utamanya adalah untuk membangun jalur komunikasi terenkripsi untuk tujuan pengujian internal, memastikan aplikasi web berfungsi dengan benar di bawah protokol HTTPS sebelum diterapkan di lingkungan produksi (Chen et al., 2018). Dengan demikian, self-signed certificate menyediakan enkripsi fungsional tanpa memerlukan validasi domain publik yang kompleks (Bhargavan et al., 2021).

2.3 Sistem Keamanan

Dalam era digital yang terus berkembang pesat ini, keamanan informasi dalam sebuah situs web telah menjadi isu yang amat krusial (Alby et al., 2022). Bisa dibilang, ini adalah prioritas utama bagi setiap pengembang web. Hal ini dikarenakan jika aspek keamanan diabaikan, dampaknya bisa sangat merugikan, seorang peretas bisa dengan mudah mengambil data-data penting, bahkan mengubah atau merusak tampilan web sesuka hati mereka.

Pentingnya sistem keamanan tidak hanya terbatas pada lingkungan korporat atau pemerintahan, tetapi juga merambah ke ranah personal dan publik, termasuk situs web dan aplikasi online (Xia et al., 2023). Seiring dengan peningkatan ketergantungan pada teknologi digital, risiko kejahatan siber seperti peretasan, pencurian data, dan penyalahgunaan informasi juga turut meningkat (Tripathy, 2024). Oleh karena itu, membangun sistem keamanan yang kokoh bukan lagi pilihan, melainkan sebuah keharusan.

Implementasi sistem keamanan yang efektif memerlukan pemahaman mendalam tentang potensi ancaman, kerentanan sistem, serta teknologi dan strategi mitigasi yang tersedia. Hal ini melibatkan analisis risiko, perancangan arsitektur keamanan,



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

penerapan kontrol teknis dan administratif, serta pemantauan berkelanjutan untuk mendeteksi dan merespons insiden keamanan. Dengan demikian, sistem keamanan bukan hanya tentang mencegah serangan, tetapi juga tentang bagaimana sistem dapat bertahan dan pulih dari insiden yang mungkin terjadi.

2.4 Keamanan Website

Keamanan informasi dalam sebuah website menjadi sangat penting. Keamanan informasi sebuah website merupakan salah satu prioritas yang sangat utama bagi seorang web development, jika seseorang mengabaikan keamanan tersebut maka seorang hacker dapat mengambil data penting serta mengacak-acak tampilan web tersebut. Ada 3 tujuan penting tercapainya keamanan informasi (Nugroho et al., 2024):

1. Confidentiality (kerahasiaan) yaitu informasi hanya tersedia untuk orang atau sistem yang memang perlu akses ke sana.
2. Integrity (kesatuan) yaitu informasi hanya dapat ditambah atau diperbarui oleh orang yang telah diotorisasi.
3. Availability (ketersediaan) yaitu informasi harus tersedia dalam waktu yang tepat ketika dibutuhkan

2.5 Ubuntu Server

Ubuntu merupakan salah satu distribusi Linux yang berbasis Debian dan didistribusikan sebagai software bebas. Nama Ubuntu berasal dari filosofi Afrika Selatan yang berarti “Kemanusiaan kepada sesama”. Ubuntu didesain untuk kepentingan penggunaan personal, namun versi server Ubuntu juga tersedia, dan telah dipakai secara luas.

Proyek Ubuntu resmi disponsori oleh Canonical Ltd. yang merupakan sebuah perusahaan yang dimiliki oleh pengusaha Afrika Selatan Mark Shuttleworth. Tujuan dari distribusi Linux Ubuntu adalah membawa semangat yang terkandung di dalam Filosofi Ubuntu ke dalam dunia perangkat lunak.

Ubuntu adalah sistem operasi lengkap berbasis Linux, tersedia secara bebas dan mempunyai dukungan baik yang berasal dari komunitas maupun tenaga ahli profesional. Selain itu, Ubuntu juga bersifat Open Source. Open source adalah

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Kekuatan komunitas seluruh dunia yang sangat ahli terampil yang membangun, berbagi dan meningkatkan perangkat lunak yang sangat terbaru kemudian membuatnya tersedia untuk semua orang.

Dalam penelitian ini penulis akan menggunakan *ubuntu server* dengan versi 24.04. Alasan dipilihnya versi 24.04 adalah karena versi tersebut merupakan versi terbaru, dan juga versi tersebut memiliki banyak dokumentasi konfigurasi yang memudahkan penulis untuk melakukan penelitian.



Gambar 2.1 Ubuntu
(sumber:course-net.com)

2.6 Web Server

Web server adalah perangkat lunak komputer yang dirancang untuk melayani permintaan HTTP (*Hypertext Transfer Protocol*) dari client (umumnya peramban web seperti Chrome, Firefox, atau Edge) dan mengirimkan kembali sumber daya yang diminta, seperti halaman web, gambar, video, atau berkas lainnya. Peran utama web server adalah sebagai penyedia informasi atau data yang dibutuhkan oleh client untuk menampilkan sebuah situs web.

Ketika seorang pengguna memasukkan URL (Uniform Resource Locator) ke dalam peramban web, atau mengklik sebuah tautan, peramban akan mengirimkan permintaan HTTP ke alamat IP dari web server yang menghosting situs web tersebut. Web server kemudian akan memproses permintaan ini, mencari berkas yang diminta, dan jika ditemukan, akan mengirimkan kembali berkas tersebut kepada client melalui protokol HTTP. Proses ini terjadi secara bolak-balik antara client sebagai pihak yang meminta informasi dan server sebagai pihak yang menyediakan informasi.

Selain melayani berkas statis (seperti berkas HTML, CSS, dan JavaScript), banyak web server modern juga mampu menangani konten dinamis. Untuk konten dinamis,

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

web server akan berinteraksi dengan program atau script lain (misalnya yang ditulis dalam PHP, Python, atau Ruby) yang berjalan di sisi server untuk menghasilkan halaman web yang disesuaikan sebelum dikirimkan ke client.



Gambar 2.2 Web Server
(sumber:cleanpng.com)

2.7 Apache Web Server

Apache HTTP Server, atau yang lebih dikenal dengan Apache, adalah salah satu perangkat lunak *web server* sumber terbuka yang paling populer dan banyak digunakan di seluruh dunia (Koch & Hao, 2021). Dikembangkan oleh Apache Software Foundation, perangkat lunak ini memiliki peran krusial dalam ekosistem internet sebagai fondasi bagi jutaan situs web. Fungsi utamanya adalah menerima permintaan HTTP (*Hypertext Transfer Protocol*) dari *client* (seperti *web browser*) dan mengirimkan kembali konten web yang diminta, seperti halaman HTML, gambar, *stylesheet*, dan *script*.

Sebagai *web server*, Apache tidak hanya bertugas melayani konten statis, tetapi juga mampu berintegrasi dengan berbagai modul dan bahasa pemrograman *server-side* seperti PHP, Python, dan Perl, untuk melayani konten dinamis. Fleksibilitas ini memungkinkan Apache untuk mendukung berbagai aplikasi web yang kompleks dan bervariasi. Arsitekturnya yang modular memungkinkan administrator untuk mengonfigurasi fitur-fitur tertentu sesuai kebutuhan, seperti *virtual hosting* (menjalankan beberapa domain di satu *server*), *load balancing*, dan *reverse proxy*.

Salah satu kekuatan utama Apache terletak pada kemampuannya untuk beroperasi di berbagai sistem operasi, termasuk Linux (seperti Ubuntu yang sering digunakan dalam konteks server), Windows, macOS, dan sistem operasi berbasis Unix lainnya (Poghosyan, 2024). Dukungan lintas platform ini, ditambah dengan sifatnya yang

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

open source, menjadikannya pilihan yang sangat atraktif bagi individu maupun organisasi dari skala kecil hingga besar.

Meskipun memiliki keunggulan dalam hal fleksibilitas dan adaptabilitas, *web server* seperti Apache seringkali menjadi target berbagai jenis serangan siber, baik yang bersifat minor maupun mayor, yang dapat berakibat fatal. Ancaman ini bervariasi mulai dari upaya penyusupan yang bertujuan untuk mengambil data penting, hingga serangan perusakan yang dapat mengganggu ketersediaan layanan. Oleh karena itu, memastikan keamanan Apache Web Server merupakan tugas esensial bagi seorang administrator sistem. Ini melibatkan penerapan konfigurasi yang aman, pembaruan perangkat lunak secara berkala, serta integrasi dengan mekanisme keamanan tambahan seperti SSL/TLS untuk enkripsi komunikasi dan *firewall* aplikasi web.



Gambar 2.3 Apache Web Server
(sumber: <http://httpd.apache.org>)

2.8 Webmin

Webmin merupakan control panel untuk mengatur perangkat-perangkat server di linux. Dengan menggunakan webmin dapat mengatur semua aspek di mesin linux secara lebih terintegrasi (Westfall, 2021).

Webmin sebagian besar dibangun menggunakan perl, dan berjalan sendiri menggunakan web server. Port Standar yang digunakan pada webmin adalah port 10000. Karena menggunakan perl maka akan mempermudah untuk menambahkan fungsi baru. Hal ini membuat pekerjaan administrasi sistem jauh lebih mudah.

Webmin juga memungkinkan untuk mengendalikan banyak server dari satu buah webmin pada subnet yang sama atau menggunakan LAN. Namun webmin memiliki kekurangan tidak ada dalam official repository dari debian dan Ubuntu karena tidak compatible dengan cara handle packages kedua sistem operasi tersebut.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Gambar 2.4 Webmin

(sumber: syafuddinanwar.web.id)

2.9 Self-signed CA

Teknologi keamanan data pada komunikasi web menjadi krusial seiring dengan meningkatnya ancaman siber. Salah satu pilar utama dalam mengamankan komunikasi online adalah penggunaan Secure Socket Layer (SSL) dan Transport Layer Security (TLS). SSL/TLS merupakan protokol keamanan digital yang memungkinkan proses komunikasi terenkripsi antara web browser pengguna dan web server. Dengan adanya sertifikat SSL/TLS, alamat URL suatu website akan berubah dari 'HTTP' menjadi 'HTTPS', di mana huruf 'S' mengindikasikan bahwa data yang ditransmisikan telah dienkripsi dan terlindungi dari penyadapan atau manipulasi oleh pihak yang tidak berwenang (Aayush et al., 2022).

Dalam konteks pengembangan dan pengujian sistem di lingkungan lokal atau terisolasi, kebutuhan akan sertifikat SSL/TLS tetap ada untuk mensimulasikan kondisi produksi dan memverifikasi fungsionalitas enkripsi. Untuk tujuan ini, Self-Signed Certificate Authority (CA) menjadi solusi yang relevan dan praktis. Berbeda dengan sertifikat yang diterbitkan oleh CA publik yang diakui secara global (seperti Self-signed CA), sertifikat self-signed adalah sertifikat yang ditandatangani oleh pembuatnya sendiri. Ini berarti entitas yang menerbitkan dan entitas yang menerima sertifikat adalah sama. Meskipun peramban web akan menampilkan peringatan keamanan "*Not secure*" atau ikon gembok merah karena sertifikat ini tidak diakui oleh root CA publik yang secara default dipercaya oleh peramban, penggunaan self-signed CA memungkinkan pengaktifan dan pengujian protokol HTTPS secara penuh di lingkungan localhost atau jaringan internal yang tidak memiliki domain publik yang tervalidasi (Mai et al., 2022).

Fungsi utama dari sertifikat self-signed CA dalam lingkungan lokal adalah untuk membangun jalur komunikasi terenkripsi antara web server dan web browser

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

pengguna untuk tujuan internal. Meskipun tidak memberikan validasi identitas seperti sertifikat publik, self-signed SSL/TLS tetap menyediakan kerahasiaan data (enkripsi) dan integritas data selama transmisi, melindungi informasi sensitif seperti kredensial login atau data uji lainnya dari penyadapan dalam jaringan lokal. Ini sangat penting untuk memastikan bahwa aplikasi web berfungsi dengan benar di bawah protokol HTTPS dan untuk menganalisis perilaku sistem keamanan sebelum diterapkan pada lingkungan produksi dengan sertifikat publik (Chen et al., 2018). Proses penerbitan sertifikat oleh Self-signed CA dilakukan melalui protokol ACME (Automated Certificate Management Environment). Protokol ini memungkinkan *web server* untuk berinteraksi dengan CA Self-signed CA untuk secara otomatis membuktikan kontrol atas domain, serta untuk mendapatkan, memperbarui, dan mencabut sertifikat (Bhargavan et al., 2021). Otomatisasi ini menghilangkan kebutuhan akan interaksi manual yang seringkali menjadi sumber kesalahan dan penundaan.

Proses pembuatan sertifikat self-signed biasanya melibatkan penggunaan tool seperti OpenSSL untuk membuat kunci pribadi dan sertifikat CA sendiri, kemudian menggunakan CA tersebut untuk menandatangani sertifikat domain yang spesifik untuk lingkungan lokal. Fleksibilitas ini menghilangkan kebutuhan akan proses validasi domain yang kompleks dan berbayar yang terkait dengan CA publik, menjadikannya pilihan yang efisien untuk kebutuhan pengembangan dan pengujian internal. Dengan demikian, self-signed CA berperan penting dalam memungkinkan analisis mendalam terhadap implementasi SSL/TLS dan ketahanan sistem di lingkungan yang terkontrol dan terisolasi.

2.10 Perbedaan HTTP dengan HTTPS

HTTP (*Hypertext transfer protocol*) adalah sebuah protokol jaringan lapisan aplikasi yang digunakan untuk sistem informasi terdistribusi, kolaboratif, dan menggunakan hipermedia (King & Wang, 2021). atau secara mudahnya HTTP merupakan aturan dasar atau protokol yang mengatur komunikasi antara client dan server (Fattah, 2021). client adalah user yang meminta dan menerima informasi sedangkan server adalah yang memberikan informasi tersebut (Nyabuto, 2023). HTTPS (*Hypertext transfer protocol*) merupakan versi aman dari HTTP, tugasnya

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

sama persis namun dalam pertukaran data https menggunakan autentikasi dan komunikasi tersandi. HTTPS di kembangkan oleh *Netscape Communications Corp.*

Perbedaan yang paling mencolok antara HTTP dan HTTPS adalah cara kerja dalam pengiriman data. HTTP tidak memberikan pengamanan dalam komunikasi antara client dan server sedangkan HTTPS menambahkan pengamanan lebih dimana client dan server dapat berkomunikasi dengan aman (Hu et al., 2021). Dalam pengamanan data antara client dan server, komunikasi data yang dikirimkan akan terenkripsi. enkripsi tersebut berada pada Socket Layer terenkripsi SSL (*Secure*) atau *Transport Layer Security* (TLS) sehingga walaupun data tercuri, data tersebut tidak dapat dibaca karena terenkripsi (Khan et al., 2022).



Gambar 2.5 HTTP dan HTTPS
(sumber: bse.telkomuniversity.ac.id)

2.11 Man-In-the-Middle

Serangan Man-in-the-Middle (MITM) adalah sebuah metode serangan siber di mana penyerang secara diam-diam menyadap dan berpotensi memanipulasi komunikasi antara dua pihak, misalnya antara client dan server (Zhang et al., 2021). Dalam konteks penelitian ini, simulasi serangan MITM dilakukan secara spesifik untuk menguji dan mengevaluasi efektivitas enkripsi SSL/TLS dalam memitigasi ancaman penyadapan data. Pengujian ini bertujuan untuk membandingkan secara langsung tingkat keamanan koneksi sebelum (menggunakan HTTP) dan sesudah (menggunakan HTTPS) implementasi sertifikat keamanan (Kponyo et al., 2020).

Pada koneksi yang tidak terenkripsi seperti HTTP, serangan MITM sangatlah berbahaya. Ketika serangan ini berhasil dilakukan, seluruh lalu lintas data yang



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

ditransmisikan antara client dan server dapat disadap dan dianalisis oleh penyerang dalam bentuk plaintext atau teks biasa (Kampourakis et al., 2022). Hal ini membuat informasi sensitif, termasuk kredensial seperti username dan password, menjadi sangat rentan untuk dicuri karena dapat dibaca secara langsung oleh penyerang menggunakan alat bantu seperti Ettercap.

Implementasi SSL/TLS yang mengubah koneksi menjadi HTTPS terbukti menjadi mitigasi yang efektif terhadap serangan MITM. Meskipun penyerang masih dapat menempatkan diri di tengah-tengah jalur komunikasi, data yang berhasil disadap tidak lagi dalam bentuk plaintext. Sebaliknya, data tersebut sudah terenkripsi dan hanya menampilkan ciphertext yang tidak dapat diinterpretasikan secara langsung oleh penyerang (Al-Shareeda et al., 2020). Dengan demikian, kerahasiaan dan integritas data tetap terjaga, dan informasi sensitif seperti kredensial pengguna tidak akan terekam oleh pihak yang tidak berwenang.

Untuk melakukan simulasi serangan MITM dalam sebuah lingkungan pengujian yang terkontrol, seringkali digunakan perangkat lunak khusus. Dalam penelitian ini, sistem operasi Kali Linux digunakan sebagai platform penyerang karena menyediakan berbagai alat bawaan untuk audit keamanan. Alat utama yang digunakan untuk serangan MITM adalah Ettercap, yang dipilih karena kemampuannya yang memadai untuk melakukan penyerangan, khususnya melalui Teknik ARP spoofing. Selanjutnya, alat seperti Wireshark dan Network Miner dimanfaatkan untuk menangkap dan menganalisis lalu lintas jaringan guna memverifikasi apakah data benar-benar terenkripsi selama simulasi serangan berlangsung (Mallik, 2019).



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

BAB III METODE PENELITIAN

3.1 Rancangan Penelitian

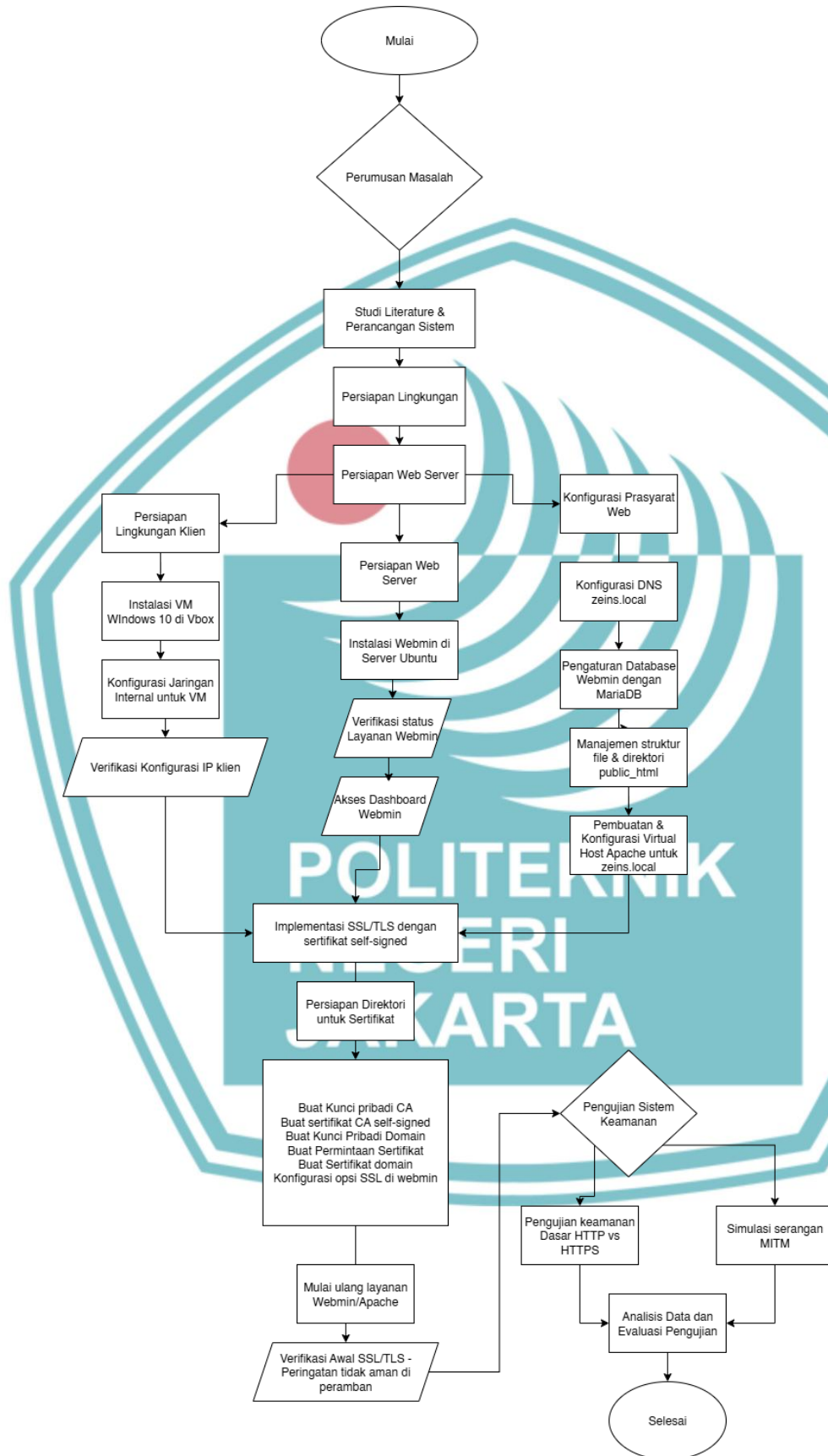
Dalam merancang penelitian, penggunaan jenis penelitian yang tepat dapat membantu peneliti untuk memperoleh gambaran yang jelas mengenai tujuan yang ingin dicapai serta masalah yang dihadapi, dengan melakukan langkah-langkah dalam mengatasi masalah tersebut. Jenis penelitian yang digunakan dalam penelitian ini adalah *realistic*, yang mana pemecahan masalah dilakukan secara real, sebagai pengembangan kerangka teori kasus untuk memperoleh verifikasi serta penilaian dalam bentuk dukungan data bagi kasus di lapangan dalam konteks teknologi dan industri. Software yang digunakan adalah ubuntu yang di dalamnya nanti akan dilakukan instalasi webmin yang kemudian nanti akan di enkripsi menggunakan *self-signed CA* setelah hal itu nantinya akan diuji coba, setelah itu menganalisa hasil dari penelitian tersebut.

3.2 Tahapan Penelitian

Dalam melakukan penelitian ini, terdapat tahapan-tahapan yang sistematis. Tahapan besar dalam penelitian ini meliputi Perumusan Masalah, Studi Literatur dan Perancangan Sistem, Persiapan Lingkungan, Implementasi SSL/TLS dengan Sertifikat Self-Signed, Pengujian Sistem Keamanan, Analisis Data dan Evaluasi Pengujian, Simpulan, dan Saran. Setiap tahapan dijelaskan secara lebih rinci dalam flowchart berikut.

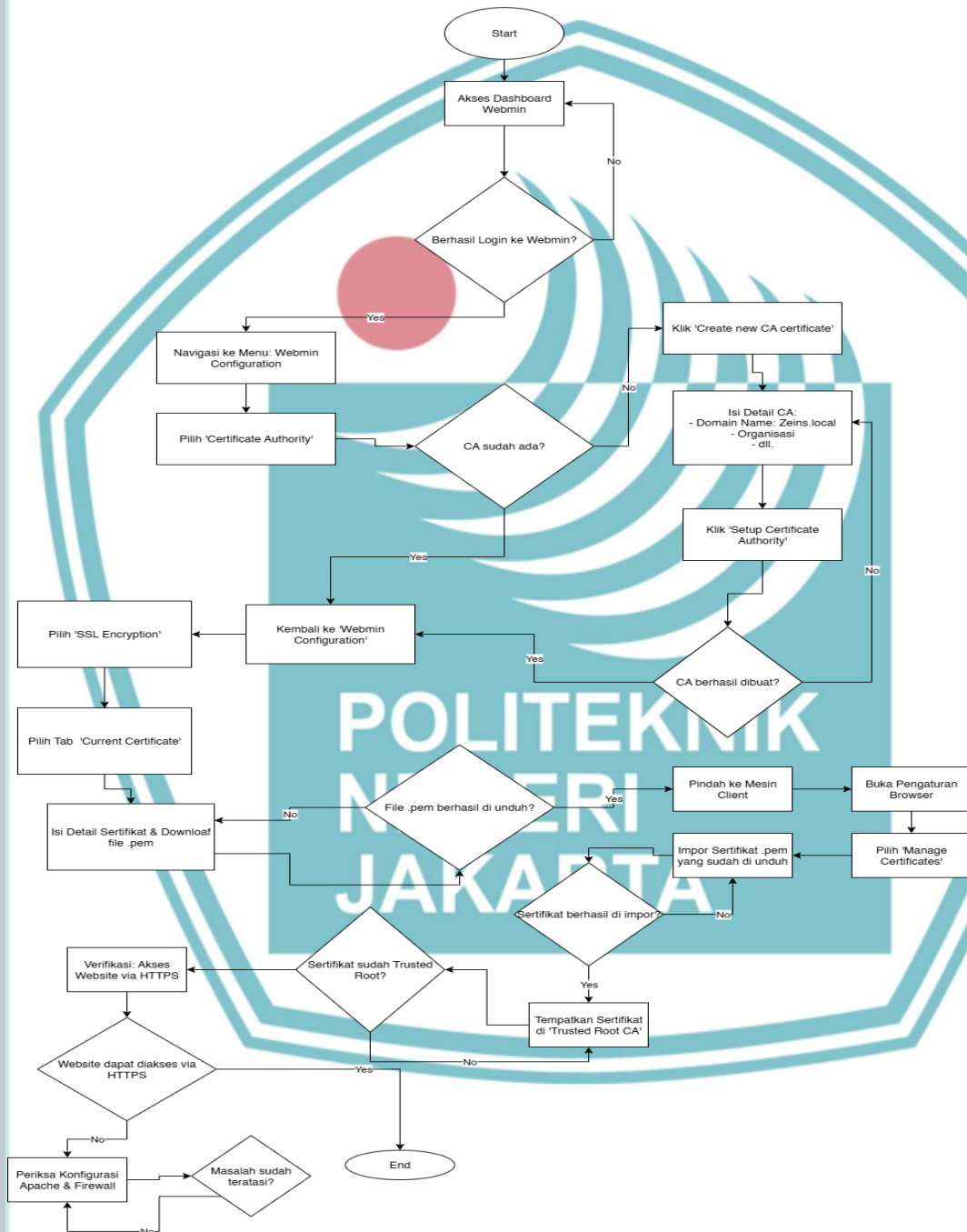
Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Gambar 3.1 Flowchart Tahapan Penelitian

Secara khusus, tahapan implementasi sistem keamanan yang mencakup pembuatan dan konfigurasi sertifikat self-signed mengikuti prosedur teknis yang dirinci pada Gambar 3.2 di bawah ini. Flowchart ini menjabarkan langkah-langkah mulai dari pembuatan Certificate Authority (CA) lokal hingga instalasi sertifikat pada client.



Gambar 3.2 Flowchart Self-signed

- Hak Cipta :**
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
 2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

3.2.1 Latar Belakang

Pada tahap ini, penulis melakukan identifikasi masalah yang ingin diselesaikan secara spesifik dengan cara merumuskan masalah dalam tiga bentuk pertanyaan, selain itu juga di definisikan lebih lanjut tentang tujuan penelitian yang di dapatkan dari rumusan masalah penelitian.

3.2.2 Observasi

Pada tahap ini, penulis melakukan studi literatur dari penelitian yang sudah ada, jurnal, artikel, buku, dan dokumen legak sebagai bahan pendukung dalam penelitian.

3.2.3 Perancangan

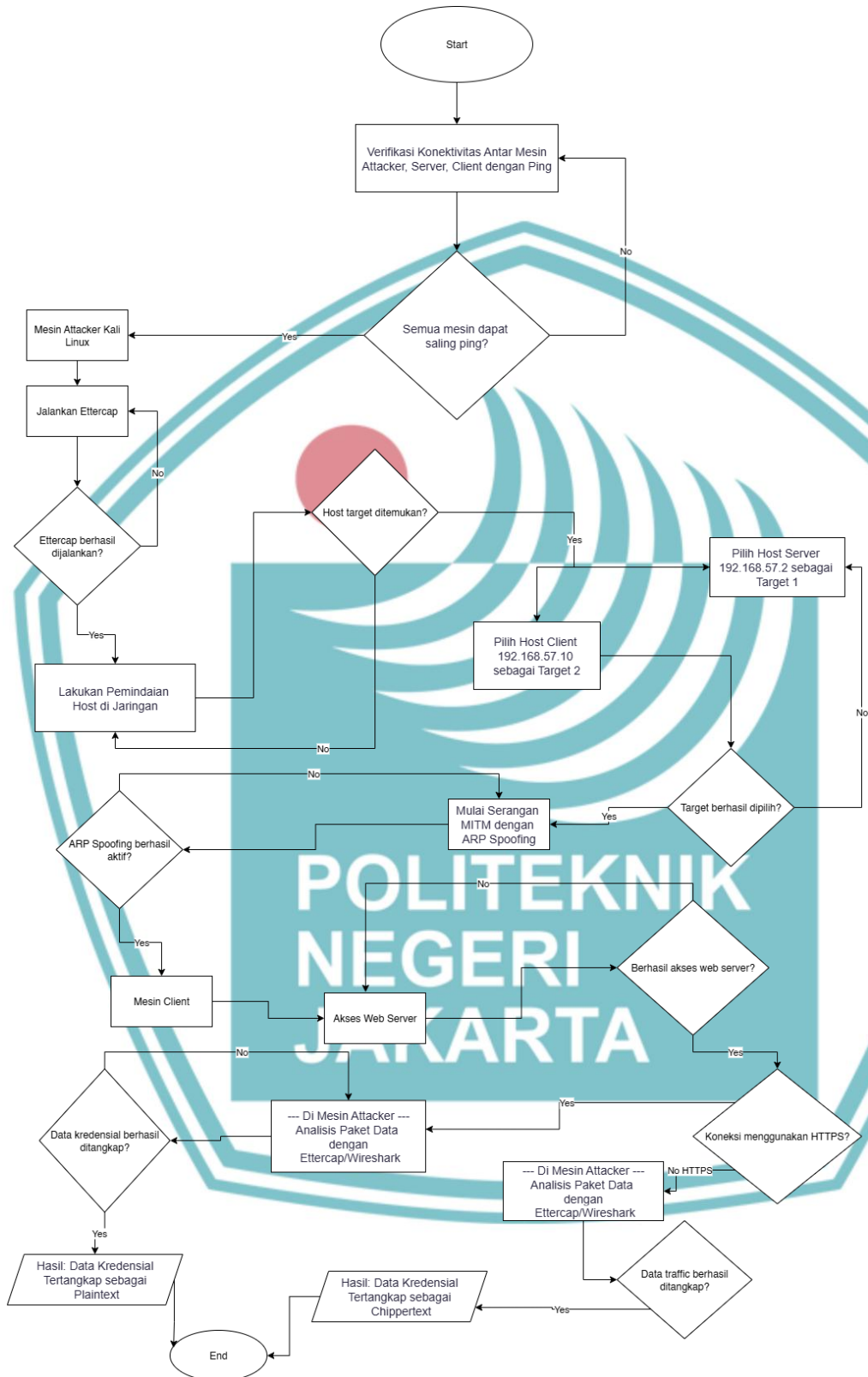
Setelah memperoleh data dari tahap observasi, dilakukanlah perancangan proses atau skema penerapan *web server* tersebut pada tools self-signed ca dan perancangan skema pengujian.

3.2.4 Pengujian

Pada tahap ini dilakukannya proses uji coba terhadap domain yang belum dan akan dienkripsi serta dilakukan analisa oleh penulis. Jika kemungkinan terdapat kendala maka akan dilakukan evaluasi kembali. Prosedur pengujian ketahanan sistem terhadap serangan *Man-in-the-Middle* (MITM) dilakukan secara sistematis seperti yang digambarkan pada Gambar 3.3.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Gambar 3.3 Flowchart cara kerja Man-in-the-middle



3.3 Objek Penelitian

Hal yang menjadi sasaran penelitian ini adalah menganalisa keamanan *web server apache* menggunakan *Self-signed CA* adapun detail pelaksanaan penelitian dijabarkan sebagai berikut:

- a. Judul Penelitian : Analysis sistem keamanan pada apache web server menggunakan Self-signed CA di webmin
- b. Konteks Penelitian : Keamanan web server apache
- c. Tempat Penelitian : Laptop Asus AMD Ram 16GB Virtual Box
- d. Waktu Penelitian : 20 Juni – 29 Juli
- e. Pihak Terkait : -

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta





Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

BAB IV HASIL DAN PEMBAHASAN

4.1 Analisis Kebutuhan

Analisis kebutuhan merupakan tahapan awal dalam fase perancangan sistem yang bertujuan untuk mengidentifikasi secara komprehensif seluruh persyaratan fungsional dan non-fungsional yang diperlukan untuk mencapai tujuan penelitian. Dalam konteks analisis sistem keamanan pada Apache Web Server menggunakan Self-signed CA di Webmin, analisis kebutuhan difokuskan pada identifikasi sumber daya, perangkat lunak, dan konfigurasi yang esensial untuk implementasi dan pengujian sistem.

Berdasarkan perumusan masalah dan tujuan penelitian, kebutuhan yang teridentifikasi meliputi:

4.1.1 Kebutuhan Perangkat Keras

Untuk menjalankan seluruh komponen sistem dan memastikan proses implementasi serta pengujian yang berjalan lancar, kami memerlukan spesifikasi perangkat keras yang memadai. Server pengujian harus memiliki kapasitas olah data (CPU), memori kerja (RAM), dan ruang penyimpanan (storage) yang cukup besar. Ini bukan hanya untuk menampung sistem operasi itu sendiri, melainkan juga untuk mengakomodasi Apache Web Server, Webmin, serta modul Self-signed CA, termasuk kebutuhan ruang untuk file-file website, log aktivitas, dan proses enkripsi-dekripsi yang intensif. Tanpa kapasitas yang memadai, kinerja sistem server bisa terhambat, bahkan mengganggu validitas hasil pengujian.

Selain itu, konektivitas jaringan yang stabil juga sangat diperlukan. Server harus selalu terhubung ke internet agar bisa mengunduh dan menginstal paket-paket perangkat lunak yang dibutuhkan, serta menjalankan proses validasi domain oleh Self-signed CA yang memerlukan koneksi ke server mereka, dan tentu saja, memungkinkan akses dari peramban web (sebagai client) untuk serangkaian pengujian.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

4.1.2 Kebutuhan Perangkat Lunak

- a. Virtual Box (VBox): Virtual Box akan digunakan untuk menciptakan mesin virtual (VM) yang akan berfungsi sebagai web server pengujian. Virtual Box yang digunakan merupakan versi 7.1.10. Penggunaan virtualisasi memungkinkan isolasi lingkungan pengujian dari sistem operasi host utama, serta memfasilitasi replikasi dan snapshot kondisi sistem sebelum dan sesudah perubahan konfigurasi.
- b. Ubuntu Server: Ubuntu Server akan diinstal sebagai sistem operasi dasar secara virtual di Virtual Box. Ubuntu yang dipilih dalam penelitian kali ini adalah ubuntu server versi 24.04 . Ubuntu versi 24.04 dipilih karena merupakan salah satu distribusi Linux yang populer, bersifat open source, dan memiliki dukungan komunitas serta tenaga ahli profesional yang luas, sehingga memudahkan proses instalasi dan konfigurasi perangkat lunak yang dibutuhkan.
- c. Windows (*Operating System Client*): Sistem operasi Windows akan diinstal sebagai mesin virtual *client*. Windows yang akan digunakan adalah Windows 10. Windows 10 dipilih karena versi windows ini tidak terlalu berat untuk dijalankan di *virtual machine* serta memiliki fitur yang lengkap untuk melakukan penelitian. Windows 10 Ini akan digunakan untuk mensimulasikan pengguna akhir yang mengakses *web server* pengujian dan untuk melakukan observasi perilaku peramban web terhadap koneksi HTTP dan HTTPS.
- d. Kali Linux (*Operating System Attacker*): Kali Linux akan diinstal sebagai mesin virtual *attacker*. Kali linux yang akan digunakan pada penelitian ini adalah Kali Linux versi 2025.2 Changelog. Distribusi Linux ini dipilih karena secara khusus dirancang untuk pengujian penetrasi dan audit keamanan, menyediakan berbagai *tools* bawaan yang essensial untuk simulasi serangan siber seperti *Man-in-the-Middle* (MITM).
- e. Web Server (*Apache HTTP Server*): Apache HTTP Server adalah perangkat lunak web server utama yang akan diinstal pada Ubuntu. Apache akan bertanggung jawab untuk melayani permintaan HTTP



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

dan HTTPS, serta akan menjadi platform di mana implementasi sistem keamanan dan enkripsi akan dilakukan.

- f. Webmin (*Control Panel*): Webmin akan diinstal sebagai panel kontrol berbasis web. Fungsi Webmin adalah untuk mempermudah administrasi dan konfigurasi perangkat-perangkat server di Linux, termasuk pengelolaan Apache Web Server dan integrasi dengan Self-signed CA. Tujuan penggunaan Webmin adalah untuk menyederhanakan tugas administrasi sistem secara terintegrasi.
- g. SSL (*Secure Sockets Layer*): Meskipun Self-signed CA menyediakan sertifikat, pemahaman tentang SSL/TLS itu sendiri adalah kebutuhan konseptual dan operasional. SSL/TLS adalah sebuah protokol keamanan digital yang memungkinkan komunikasi terenkripsi antara web browser dan web server. Penerapan SSL/TLS (yang akan disediakan oleh Self-signed CA) akan mengubah protokol HTTP menjadi HTTPS, dan memastikan privasi data.
- h. OpenSSL: Sebagai library kriptografi open-source yang tersedia pada sistem operasi berbasis Linux (seperti Ubuntu dan Kali Linux), OpenSSL akan digunakan untuk pembuatan sertifikat self-signed (kunci pribadi dan sertifikat CA lokal) yang diperlukan untuk mengaktifkan HTTPS di lingkungan pengujian virtual.
- i. Self-signed CA: Self-signed CA adalah *Certificate Authority* (CA) utama yang akan digunakan untuk mendapatkan dan mengimplementasikan sertifikat SSL/TLS secara gratis pada Apache Web Server. Tool ini berperan krusial dalam mengubah URL situs web menjadi 'HTTPS' dan mengamankan komunikasi data melalui enkripsi.
- j. Ettercap: Ettercap digunakan Sebagai alat bantu pada Kali Linux, Ettercap akan digunakan untuk simulasi serangan Man-in-the-Middle (MITM), alasan memilih ettercap sebagai alat simulasi MITM dikarenakan ettercap merupakan tools yang memadai dalam melakukan penyerangan MITM, khususnya untuk melakukan ARP

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

spoofing dan mengobservasi lalu lintas jaringan dalam kondisi terenkripsi maupun tidak terenkripsi.

- k. Wireshark: Wireshark adalah penganalisis protokol jaringan yang akan digunakan untuk menangkap dan menganalisis paket data secara detail dalam berbagai skenario pengujian. Ini krusial untuk memverifikasi apakah data benar-benar terenkripsi atau dapat dibaca dalam *plaintext* selama serangan atau komunikasi normal.
- l. Network Miner: Network Miner adalah *forensic analysis tool* yang akan digunakan untuk analisis *network traffic* secara pasif, membantu mengidentifikasi *host*, sesi, dan file yang ditransfer dari *packet capture* yang dilakukan oleh Wireshark, memberikan wawasan tambahan mengenai lalu lintas yang diamati. Alasan kenapa Network Miner dipilih sebagai *forensic analysis tool* adalah karena Network Miner tersedia secara gratis dan mudah digunakan karena merupakan aplikasi windows berbasis GUI.

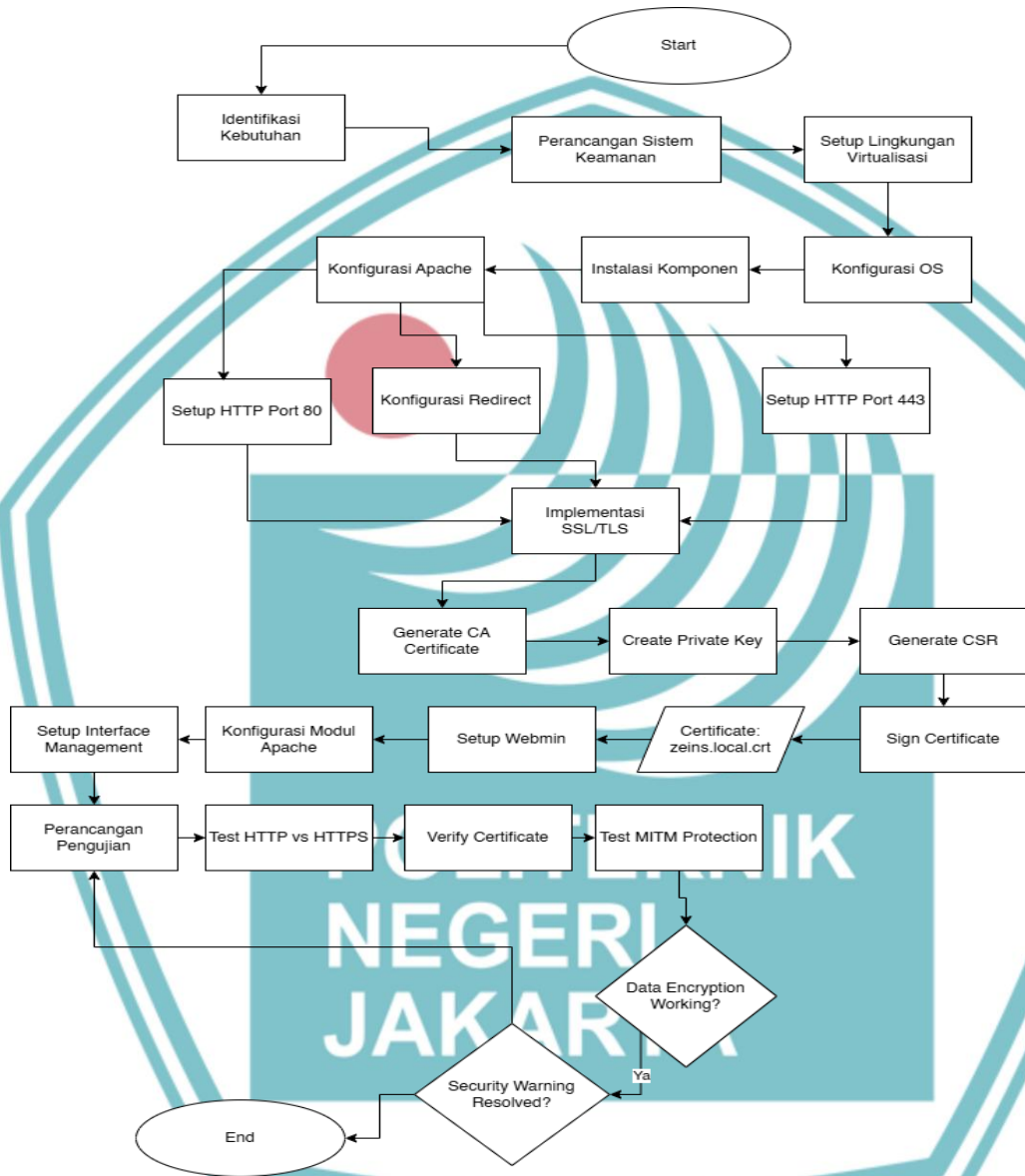
4.1.3 Kebutuhan Konfigurasi

- a. *Domain Name*: Diperlukan satu atau lebih domain name yang valid dan dapat diakses publik untuk proses validasi oleh Self-signed CA dan pengujian sistem keamanan.
- b. *DNS Management*: Konfigurasi DNS yang benar diperlukan agar domain name mengarah ke alamat IP web server yang sedang diuji.
- c. *Firewall Rules*: Pengaturan firewall yang tepat diperlukan untuk memastikan port 80 (HTTP) dan 443 (HTTPS) terbuka agar web server dapat diakses dari luar dan Self-signed CA dapat melakukan validasi domain.

4.2 Perancangan Sistem Keamanan

Perancangan sistem keamanan merupakan tahapan yang penting guna menerjemahkan hasil analisis kebutuhan untuk menjadi sebuah cetak biru sebuah arsitektur dan konfigurasi yang terstruktur. Tujuan dari perancangan ini adalah untuk menciptakan sebuah lingkungan *web server* yang terlindungi, khususnya

Apache Web Server, melalui implementasi sertifikat SSL/TLS dari *Self-signed CA* yang dikelola via Webmin. Perancangan ini berupaya memastikan terpenuhinya prinsip-prinsip keamanan seperti kerahasiaan, integritas, dan ketersediaan data.



Gambar 4.1 Flowchart Perancangan Sistem Keamanan Web Server

Adapun elemen-elemen kunci yang perlu dirancang dalam sistem keamanan ini meliputi:

- Hak Cipta :**
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
 2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

4.2.1 Arsitektur Sistem

Arsitektur sistem yang diperlukan dalam membangun sistem keamanan antara lain terdiri dari :

- a. Lingkungan Virtualisasi: Sistem akan dirancang untuk beroperasi di dalam lingkungan mesin virtual menggunakan Virtual Box. Pendekatan ini dipilih untuk memungkinkan isolasi lingkungan pengujian dari sistem *host* utama, serta memfasilitasi proses replikasi dan *snapshot* kondisi sistem yang krusial untuk analisis komparatif sebelum dan sesudah implementasi. Lingkungan virtual ini akan menampung tidak hanya *web server* utama, tetapi juga mesin *client* dan mesin *attacker*.
- b. Sistem Operasi Server: Ubuntu akan diinstal sebagai sistem operasi dasar pada mesin virtual. Pemilihan Ubuntu didasarkan pada karakteristiknya sebagai distribusi Linux open source yang populer, didukung oleh komunitas luas dan tenaga ahli profesional, yang mempermudah proses instalasi serta konfigurasi perangkat lunak lain yang dibutuhkan dalam penelitian ini.
- c. Sistem Operasi Klien dan Penyerang: Selain sistem operasi server, perancangan arsitektur juga mencakup penyediaan mesin virtual dengan Windows sebagai sistem operasi *client* untuk simulasi pengguna akhir, dan Kali Linux sebagai sistem operasi *attacker* yang akan digunakan untuk melaksanakan simulasi serangan siber (MITM). Kedua sistem operasi ini akan diinstal di dalam lingkungan Virtual Box yang sama untuk menciptakan skenario pengujian yang terkontrol dan realistis.
- d. Penempatan Komponen: Penempatan Apache Web Server, Webmin, dan *client* Self-signed CA (misalnya Certbot) dirancang secara terintegrasi di dalam lingkungan Ubuntu virtual. Sementara itu, alat-alat pengujian keamanan seperti Ettercap, Wireshark, dan Network Miner akan ditempatkan di dalam mesin virtual Kali Linux. Hal ini bertujuan untuk memastikan interaksi yang optimal antar komponen, efisiensi dalam pengelolaan sistem, dan kemampuan untuk melakukan pengujian keamanan dari perspektif *client* maupun *attacker* dalam lingkungan yang



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

terkontrol. Website webmin berfungsi sebagai aplikasi web target yang akan diamankan dan diuji.

4.2.2 Perancangan Konfigurasi Apache Web Server

Perancangan konfigurasi *Apache Web Server* terdiri dari:

- a. Konfigurasi Dasar HTTP: Apache akan dikonfigurasi awal untuk melayani permintaan HTTP standar pada port 80. Konfigurasi ini menjadi baseline sebelum web server dialihkan ke HTTPS.
- b. Persiapan untuk HTTPS: Perancangan mencakup penyiapan konfigurasi Apache untuk menerima koneksi HTTPS pada port 443. Ini melibatkan pembuatan virtual host khusus untuk domain yang akan diamankan dengan sertifikat SSL/TLS, memastikan Apache siap memproses koneksi terenkripsi.
- c. Pengalihan HTTP ke HTTPS: Aspek krusial dalam perancangan keamanan adalah implementasi pengalihan (redirect) permanen dari HTTP ke HTTPS. Ini dirancang untuk secara otomatis mengarahkan semua permintaan yang masuk melalui port 80 ke port 443, memastikan bahwa seluruh sesi komunikasi data terenkripsi sejak awal dan tidak ada informasi yang dikirimkan tanpa perlindungan.

4.2.3 Perancangan Implementasi Sertifikat SSL/TLS (Self-Signed):

Perancangan ini fokus pada metode implementasi sertifikat SSL/TLS menggunakan OpenSSL untuk membuat self-signed certificate di lingkungan lokal. Ini meliputi pembuatan sertifikat Certificate Authority (CA) lokal, private key untuk domain zeins.local, serta Certificate Signing Request (CSR) yang kemudian akan ditandatangani oleh CA lokal untuk menghasilkan sertifikat zeins.local.crt. Desain ini tidak melibatkan interaksi dengan Certificate Authority publik.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

4.2.4 Perancangan Pengelolaan via Webmin

Dalam merancang pengelolaan sertifikat menggunakan webmin yang diperlukan antara lain:

- a. Modul Webmin untuk Apache dan SSL/TLS: Perancangan akan memanfaatkan modul-modul spesifik yang tersedia di Webmin untuk mengelola konfigurasi Apache Web Server dan memfasilitasi proses instalasi serta pengelolaan sertifikat SSL/TLS yang ditandatangani sendiri. Webmin berfungsi sebagai antarmuka terpusat yang menyederhanakan tugas administrasi dan integrasi seluruh konfigurasi keamanan.
- b. Antarmuka Pengelolaan: Desain penggunaan Webmin sebagai antarmuka grafis yang intuitif bertujuan untuk memudahkan administrator dalam memantau status keamanan, sertifikat SSL/TLS, dan pengaturan web server lainnya, tanpa perlu berinteraksi langsung dengan command line.

4.2.5 Perancangan Mekanisme Pengujian

Pada tahap perancangan mekanisme pengujian, fokus utama adalah mengembangkan skenario yang jelas untuk membandingkan kondisi keamanan web server sebelum dan sesudah implementasi SSL/TLS. Skenario ini akan mencakup pengujian akses HTTP versus HTTPS, verifikasi validitas sertifikat, dan pengujian ketahanan terhadap serangan *Man-in-the-Middle* (MITM) yang akan dilaksanakan di lingkungan virtual.

Penentuan metrik keberhasilan pengujian difokuskan pada observasi perilaku data dalam lingkungan virtual. Untuk pengujian dasar HTTP/HTTPS, metrik keberhasilan diukur dari perubahan indikator pada peramban (misalnya, dari "Not Secure" menjadi ikon gembok terkunci) dan kemampuan akses situs web melalui HTTPS. Lebih spesifik, untuk pengujian serangan *Man-in-the-Middle* (MITM) yang melibatkan alat seperti Ettercap, Wireshark, dan Network Miner, metrik keberhasilan utama adalah apakah kredensial (username dan password) atau data sensitif lainnya dapat terekam dalam bentuk plaintext oleh Network Miner atau *Wireshark*. Apabila sertifikat SSL berhasil diimplementasikan dan berfungsi dengan baik pada koneksi HTTPS, maka kredensial dan data lain tidak akan

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

terekam dalam bentuk plaintext. Sebaliknya, data yang terekam seharusnya berupa *ciphertext* yang tidak dapat dibaca. Namun, jika sertifikat SSL tidak berhasil atau tidak diterapkan, maka kredensial dan data tersebut akan terekam dalam bentuk plaintext saat komunikasi melalui HTTP. Selain itu, ketiadaan peringatan keamanan yang signifikan dari alat-alat pengujian jaringan juga akan menjadi indikator keberhasilan dari perancangan sistem keamanan ini.

Perancangan yang komprehensif ini menjadi panduan yang kemudian akan diimplementasikan pada tahapan selanjutnya, yang kemudian akan memastikan bahwa setiap komponen sistem keamanan dapat terintegrasi dengan baik dan berfungsi sesuai harapan untuk melindungi *web server Apache*.

4.3 Implementasi Self-signed CA pada Apache via Webmin

Tahapan ini adalah tahapan realisasi langsung dari perancangan sistem keamanan yang telah dijelaskan sebelumnya. Dokumentasi secara rinci serta prosedur teknis instalasi dan konfigurasi *Self-signed CA* pada *Apache Web Server*, dengan memanfaatkan *interface* grafis *webmin* sebagai alat bantu administrasi akan di jelaskan disini. Implementasi ini bertujuan untuk mengaktifkan protokol HTTPS pada *web server* pengujian, memastikan seluruh komunikasi data terenkripsi, serta mengotomatisasi proses perolehan dan pembaruan sertifikat SSL/TLS.

Proses implementasi dilakukan dalam beberapa tahapan kunci, yang meliputi persiapan lingkungan server, client, attacker, dan instalasi Webmin, konfigurasi prasyarat web (DNS, database, Apache), hingga konfigurasi spesifik *Self-signed CA* untuk aktivasi HTTPS. Setiap langkah akan didukung dengan dokumentasi visual berupa tangkapan layar (*screenshot*) untuk memberikan kejelasan dan bukti pelaksanaannya.

Sebagai bagian esensial dari persiapan lingkungan implementasi, infrastruktur mesin virtual dirancang dan dikonfigurasi menggunakan VirtualBox untuk mengakomodasi peran server, client, dan attacker. Konfigurasi ini memungkinkan penciptaan jaringan virtual yang terisolasi dan terkontrol, di mana interaksi antar komponen sistem dapat diamati dan diuji secara komprehensif. Detail spesifik

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

mengenai peran dan sistem operasi yang digunakan pada masing-masing mesin virtual dalam lingkungan pengujian disajikan dalam Tabel 4.1 berikut.

Tabel 4.1 Infrastruktur Mesin Virtual

Mesin Virtual	IP Address	Subnetmask	Gateway
Client	192.168.57.10	255.255.255.0	192.168. 57.2
Server	192.168.57.2	255.255.255.0	192.168. 57.1
Attacker	192.168.57.5	255.255.255.0	192.168. 57.1

Dalam implementasi lingkungan pengujian, infrastruktur mesin virtual dirancang dan dikonfigurasi menggunakan VirtualBox untuk mengakomodasi peran server, client, dan attacker. Tabel 4.1 merinci konfigurasi jaringan setiap mesin virtual yang dibangun di VirtualBox, menciptakan lingkungan pengujian yang terisolasi dan terkontrol. Alokasi alamat IP ini dirancang untuk memfasilitasi komunikasi yang spesifik antar VM (Virtual Machine) serta mendukung skenario pengujian keamanan yang telah direncanakan. Mesin virtual Client memiliki IP Address 192.168.57.10, Subnetmask 255.255.255.0, dan Gateway 192.168.57.2. Mesin virtual Server dikonfigurasi dengan IP Address 192.168.57.2, Subnetmask 255.255.255.0, dan Gateway 192.168.57.1. Sementara itu, mesin virtual Attacker memiliki IP Address 192.168.57.5, Subnetmask 255.255.255.0, dan Gateway 192.168.57.1. Konfigurasi ini memungkinkan penciptaan jaringan virtual yang terisolasi dan terkontrol, di mana interaksi antar komponen sistem dapat diamati dan diuji secara komprehensif. Khususnya, *Default Gateway* client diatur ke IP server (192.168.57.2), yang krusial untuk skenario pengujian Man-in-the-Middle (MITM) di mana penyerang mungkin mencoba memanipulasi alur lalu lintas melalui server atau menyamar sebagai gateway.

Mesin virtual server (Ubuntu) dikonfigurasi dengan alamat IP 192.168.57.2 dan Subnetmask 255.255.255.0. Alamat IP ini dipilih sebagai identitas tetap dan mudah dikenali untuk web server utama yang menjalankan Apache. Sementara itu, Default Gateway server diatur ke 192.168.57.1. Dalam konteks jaringan internal, IP ini dapat merepresentasikan titik keluar ke jaringan yang lebih luas, atau sebagai

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

bagian dari skema jaringan virtual yang memfasilitasi komunikasi antar VM. Alamat IP server ini (192.168.57.2) juga secara spesifik berfungsi sebagai Default Gateway bagi mesin client, menunjukkan bahwa lalu lintas dari client yang ditujukan keluar dari subnet lokal akan melewati server. Selanjutnya, mesin virtual client (Windows) diberikan alamat IP 192.168.57.10 dengan Subnetmask 255.255.255.0. Alamat ini berada dalam segmen jaringan yang sama dengan server dan attacker, memastikan mereka dapat berkomunikasi langsung di dalam lingkungan virtual. *Default Gateway* untuk mesin client diatur ke 192.168.57.2, yang merupakan alamat IP dari mesin server. Konfigurasi ini krusial karena mengarahkan seluruh lalu lintas keluar dari client melalui server, sebuah prasyarat penting untuk skenario pengujian *Man-in-the-Middle* (MITM) di mana penyerang mungkin mencoba untuk memanipulasi alur lalu lintas melalui server atau menyamar sebagai server atau gateway. Terakhir, mesin virtual attacker (Kali Linux) dikonfigurasi dengan alamat IP 192.168.57.5 dan Subnetmask 255.255.255.0. Alamat ini juga berada di subnet yang sama, memungkinkan attacker untuk berinteraksi langsung dengan server dan client. Default Gateway untuk attacker diatur ke 192.168.57.1, sama dengan gateway yang diatur pada server. Penempatan attacker dalam subnet yang sama ini adalah elemen kunci untuk simulasi serangan MITM, karena memungkinkan attacker untuk melakukan ARP spoofing dan menyadap lalu lintas yang dimaksudkan untuk melewati gateway atau antar host dalam jaringan lokal. Pengaturan jaringan ini secara keseluruhan menciptakan lingkungan pengujian yang representatif dan terkontrol, di mana setiap peran (*server*, *client*, *attacker*) memiliki konfigurasi jaringan yang spesifik untuk menjalankan skenario implementasi dan pengujian keamanan sesuai rancangan.

4.3.1 Persiapan Lingkungan Server dan Prasyarat

Sebelum implementasi Self-signed CA secara spesifik, beberapa langkah fundamental perlu dilakukan untuk menyiapkan lingkungan server dan menginstal perangkat lunak dasar.

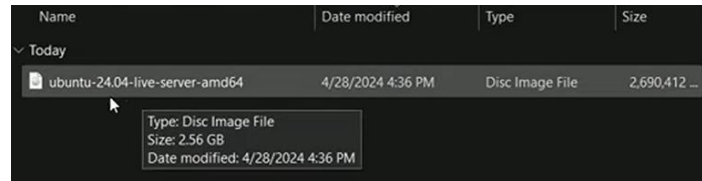
4.3.1.1 Instalasi Ubuntu Server 24.04

Untuk menginstall Ubuntu Server versi 24.04 diperlukan beberapa langkah, langkah-langkah tersebut adalah.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

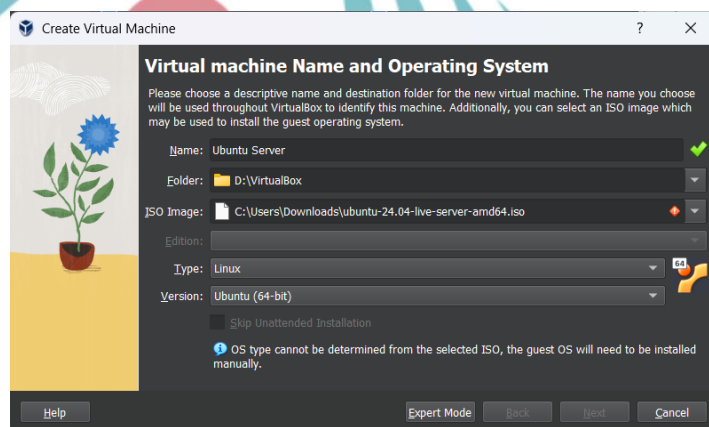
1. ISO Ubuntu Server



Gambar 4.2 File Iso Ubuntu Server

Gambar 4.2 merupakan file iso dari Ubuntu server yang berisi OS dari ubuntu server itu sendiri.

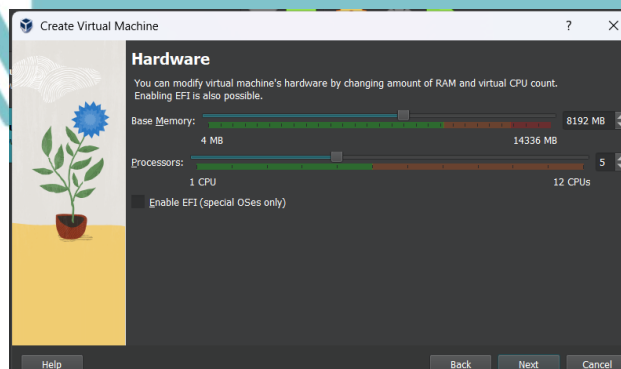
2. Create Virtual Machine Name



Gambar 4.3 Ubuntu Virtual Machine Name

Pada gambar 4.3 terlihat kolom konfigurasi yang berguna untuk memberikan nama pada sistem Ubuntu server di virtualbox.

3. Konfigurasi Hardware



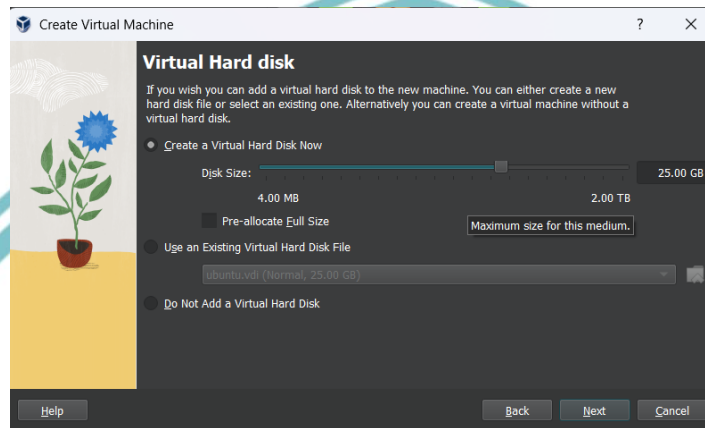
Gambar 4.4 Ubuntu Hardware Configuration

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Pada gambar 4.4 terdapat tampilan konfigurasi *hardware*. Konfigurasi ini berguna untuk mengatur alokasi *base memor* dan *processors* pada sistem Ubuntu.

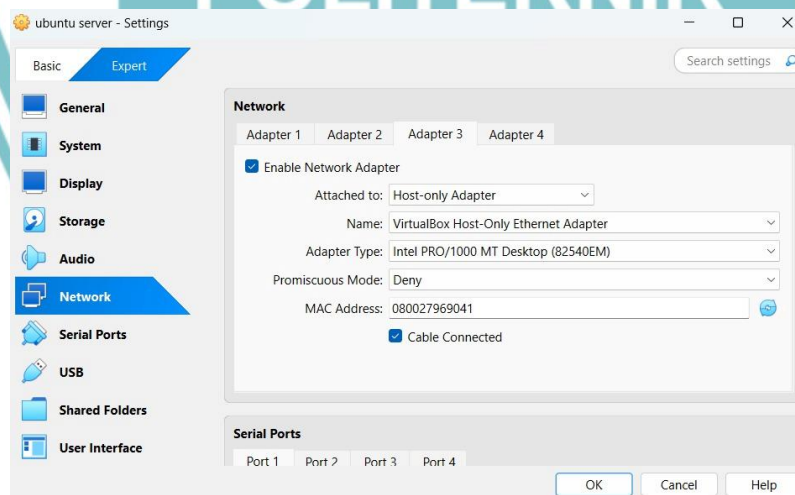
4. Virtual Hard Disk



Gambar 4.5 Ubuntu Hard disk configuration

Pada gambar 4.5 menunjukkan tampilan konfigurasi virtual hard disk. Virtual Hard Disk berfungsi sebagai kapasitas *storage* dari web server yang akan digunakan.

5. Network VBox



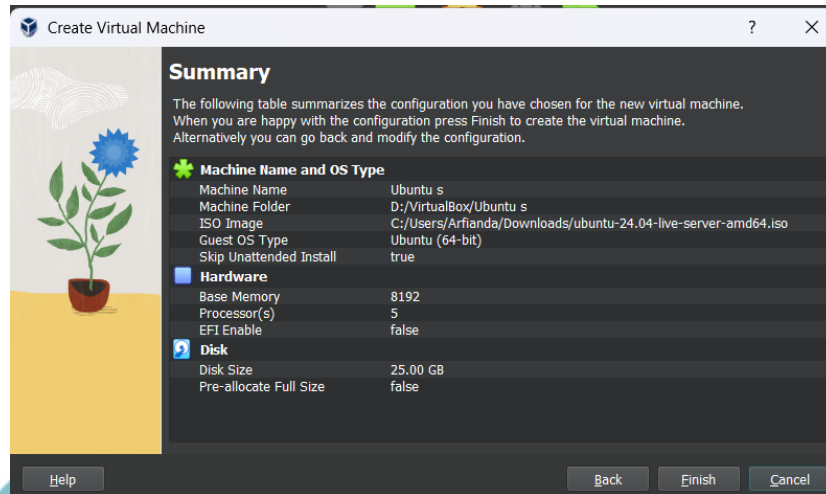
Gambar 4.6 Ubuntu Adapter Configuration

Pada gambar 4.6 diperlihatkan konfigurasi *network*. Konfigurasi ini adalah konfigurasi untuk menambahkan adapter jaringan yang akan dipasang pada Ubuntu Server.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

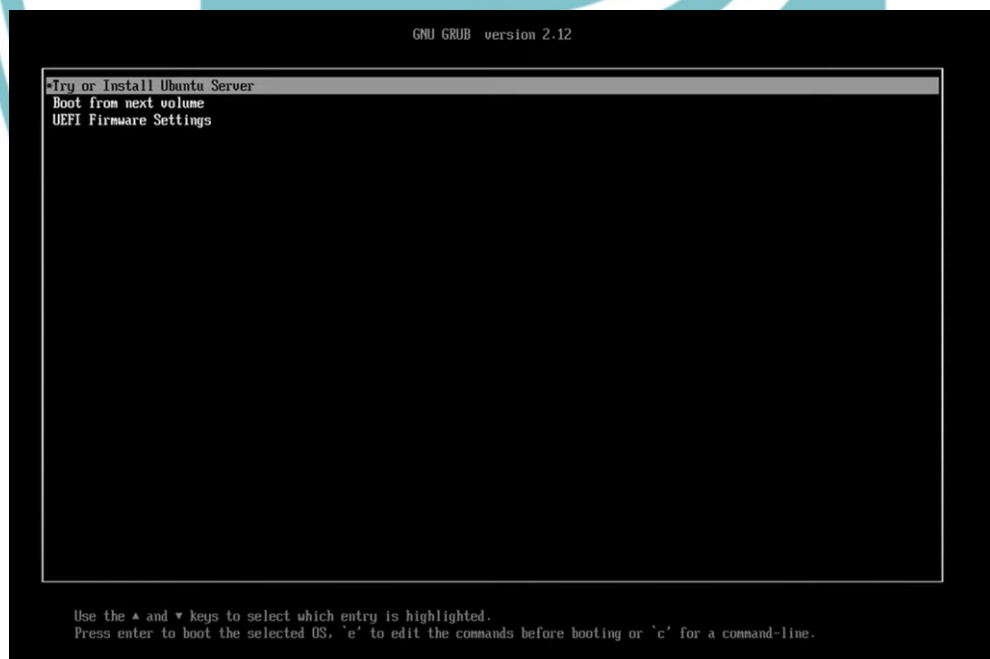
6. Summary Virtual Machine



Gambar 4.7 Ubuntu Summary

Pada gambar 4.7 di tampilkan sebuah tampilan *summary*. Bagian ini merupakan tahap terakhir dalam pembuatan Virtual Machine di virtual box.

7. GRUB Ubuntu Server



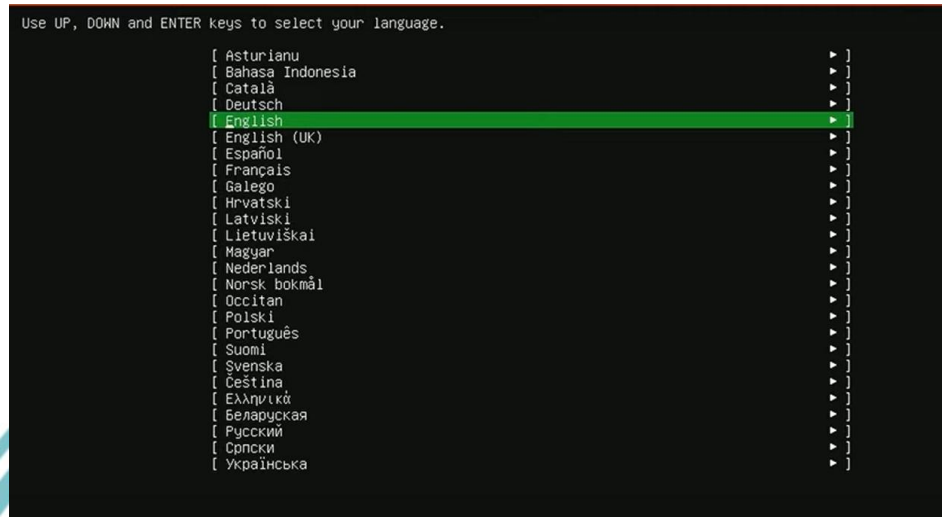
Gambar 4.8 Grub Ubuntu

Pada gambar 4.8 diperlihatkan tampilan dari tahapan GRUB. Tahapan ini merupakan tahapan awal dari instalasi iso Ubuntu Server, pada tahap ini pilih *Try or Install Ubuntu Server* untuk melakukan instalasi.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

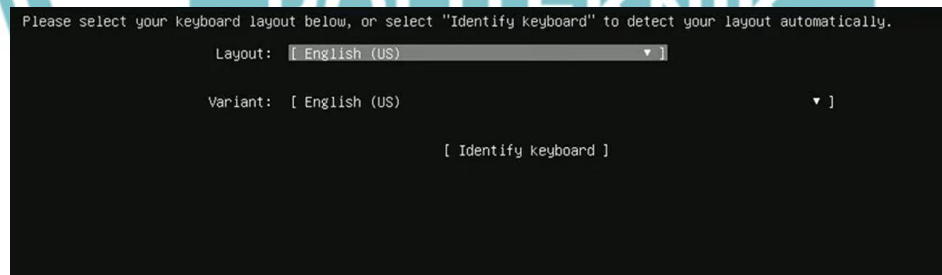
8. Language Select



Gambar 4.9 Language Setting

Pada gambar 4.9 ditampilkan tahap *language select*. Tahapan ini adalah tahapan pemilihan bahasa pada ubuntu server, bahasa dari sistem ubuntu akan menyesuaikan dengan bahasa yang dipilih.

9. Keyboard Layout



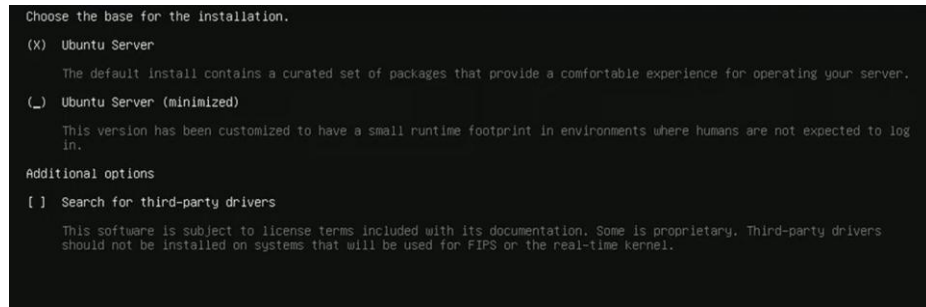
Gambar 4.10 Keyboard Layout Ubuntu

Pada gambar 4.10 diperlihatkan sebuah konfigurasi keyboard. Konfigurasi ini digunakan untuk memilih keyboard layout yang akan digunakan pada sistem Ubuntu server. Pada penelitian ini Ubuntu Server yang digunakan akan menggunakan keyboard layout *English (US)*.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

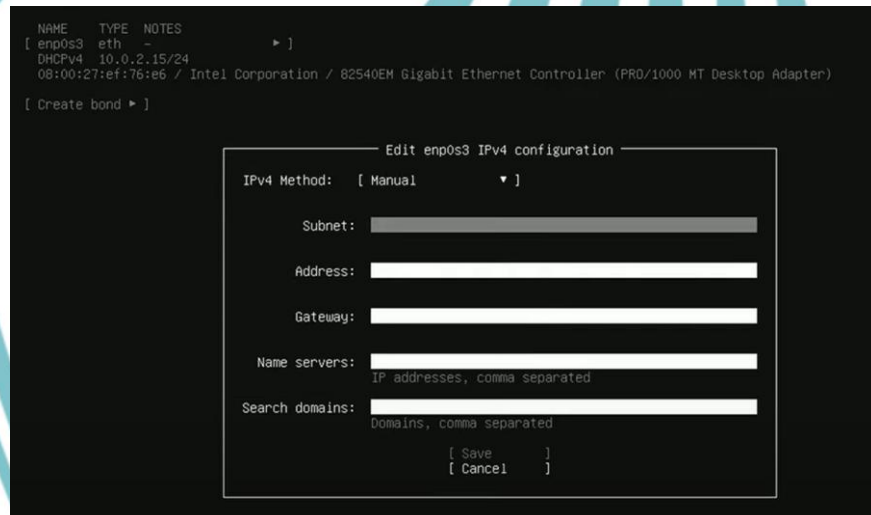
10. Base Installation



Gambar 4.11 Base Installation Ubuntu

Gambar 4.11 menampilkan tahapan *base instakkation*. Pada tahapan ini pilih Ubuntu Server sebagai *base installation*.

11. Network Configuration



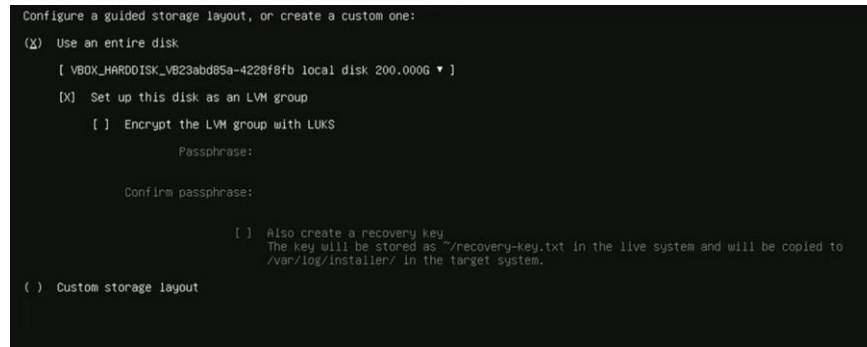
Gambar 4.12 Network Configurtion Ubuntu

Pada gambar 4.12 diperlihatkan tahapan *network configuration*. Pada *network configuration* kita akan menyesuaikan IP Address, Subnetmask dan Gateway sesuai dengan Tabel 4.1.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

12. Konfigurasi Storage



Gambar 4.13 Storage Ubuntu

Pada gambar 4.13 diperlihatkan tahapan konfigurasi *storage*. Tahapan ini merupakan tahapan untuk memilih hard disk yang akan digunakan untuk melakukan instalasi Ubuntu Server.

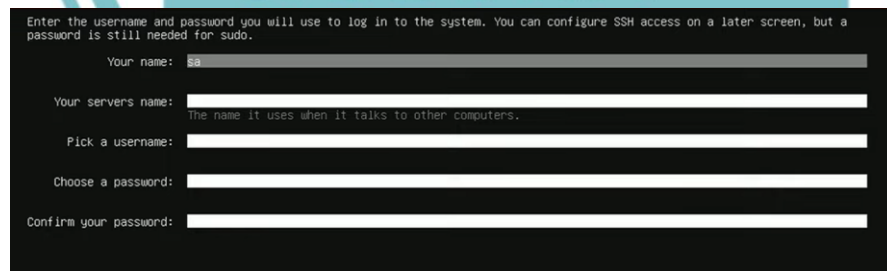
13. File System Summary



Gambar 4.14 File System Ubuntu

Pada gambar 4.14 diperlihatkan tahapan *File System Summary*. Pada tahap ini ubuntu server akan menampilkan alokasi kapasitas yang terpakai di hard disk untuk penginstalasian sistem.

14. Username setting



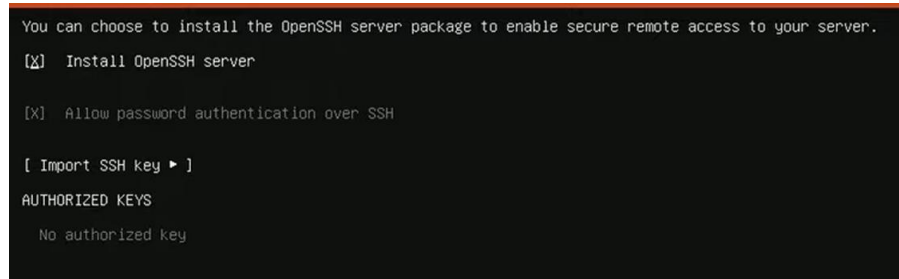
Gambar 4.15 Username Setting

Pada gambar 4.15 diperlihatkan tahapan *username setting*. Tahap ini merupakan tahap penting dimana Ubuntu Server akan meminta untuk memasukan username sudo yang akan digunakan untuk pengoperasian server.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

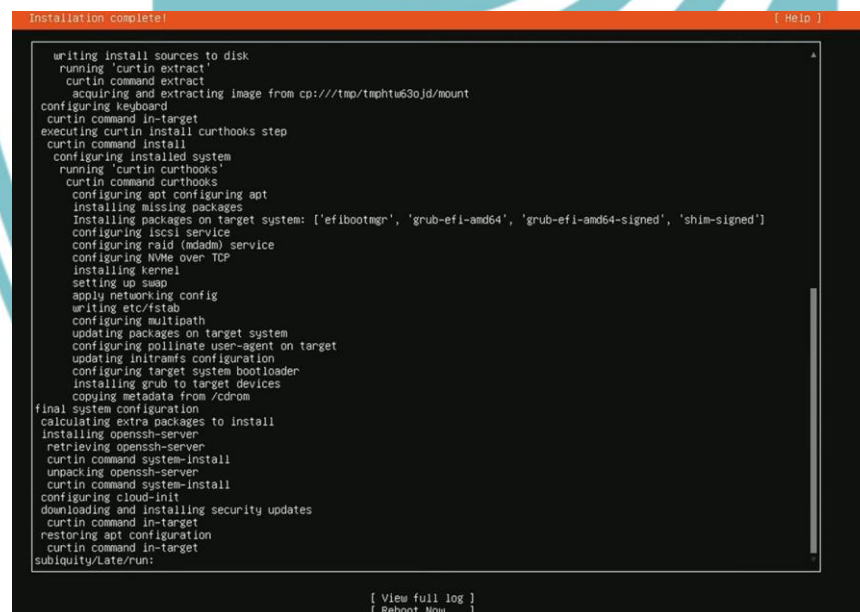
15. Install OpenSSH



Gambar 4.16 OpenSSH Ubuntu

Pada gambar 4.16 diperlihatkan tahapan *install OpenSSH*. Pada bagian ini disarankan untuk menyalakan OpenSSH server guna mempermudah untuk maintenance dan setting server secara remote.

16. Instalasi Ubuntu Server



Gambar 4.17 Instalasi Akhir Ubuntu

Pada gambar 4.17 diperlihatkan Tahap instalasi akhir ubuntu. Tahap ini merupakan tahap terakhir penginstalasian sistem. Setelah tahapan ini selesai Ubuntu server akan siap untuk digunakan.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

4.3.1.2 Pembaruan Sistem dan Instalasi Depedensi

Langkah awal adalah untuk memastikan bahwa Ubuntu dapat digunakan dan dependensi yang diperlukan terinstal. Ini mencakup *update* daftar paket dan *upgrade* paket yang ada, serta instalasi *wget* dan *apt-transport-https* untuk mengelola *repository*.

```
root@kci:~# sudo apt update
sudo: unable to resolve host kci: Name or service not known
Ign:1 http://download.webmin.com/download/repository sarge InRelease
Get:2 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Hit:3 http://download.webmin.com/download/repository sarge Release
Hit:5 http://us.archive.ubuntu.com/ubuntu focal InRelease
Get:6 http://us.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Get:7 http://us.archive.ubuntu.com/ubuntu focal-backports InRelease [108 kB]
Fetched 336 kB in 2s (143 kB/s)
Reading package lists... Done
Building dependency tree
Reading state information... Done
274 packages can be upgraded. Run 'apt list --upgradable' to see them.
```

Gambar 4.18 Update sistem ubuntu

Pada Gambar 4.18 dilakukan *update* sistem pada *ubuntu* server dengan mengetikkan perintah *sudo apt update* sehingga semua sistem di dalam *Operating System* dapat pembaruan di versi terbaru.

```
root@kci:~# apt-get upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
```

Gambar 4.19 Upgrade sistem ubuntu

Pada gambar 4.19 di lakukan upgrade sistem dengan mengetikan *apt-get upgrade*. *Apt-get upgrade* merupakan sebuah perintah untuk mengunduh dan menginstall versi terbaru dari paket yang sudah ter-install.

```
root@kci:~# apt-get install wget apt-transport-https software-properties-common
Reading package lists... Done
Building dependency tree
Reading state information... Done
software-properties-common is already the newest version (0.99.9.8).
wget is already the newest version (1.20.3-1ubuntu2).
The following NEW packages will be installed:
```

Gambar 4.20 Instalasi wget dan apt-transport-https

Perintah pada Gambar 4.20 ditujukan untuk melakukan instalasi paket *wget* yang akan digunakan untuk melakukan pengintsallan *webmin* secara otomatis.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

4.3.1.3 Penambahan Repository dan Kunci Webmin

Untuk menginstal Webmin, *repository* resminya perlu ditambahkan ke sistem Ubuntu dan kunci GPG diimpor untuk memverifikasi integritas paket dengan mengetikkan perintah seperti pada gambar 4.21 dan 4.22.

```
root@kci:~# sudo add-apt-repository "deb [arch=amd64] http://download.webmin.com/download/repository sarge contrib"
```

Gambar 4.21 Penambahan Repository Webmin

```
root@kci:~# wget -q http://www.webmin.com/jcameron-key.asc -O- | apt-key add -
```

Gambar 4.22 Impor GPG Key

4.3.1.4 Konfigurasi Firewall UFW

Pada gambar 4.23 *Firewall Uncomplicated Firewall (UFW)* dikonfigurasi untuk mengizinkan akses ke *port* 10000 TCP, yang digunakan oleh Webmin, serta *port* standar untuk HTTP (80) dan HTTPS (443) yang akan digunakan oleh Apache. Setelah aturan ditambahkan, *firewall* di-*reload* agar perubahan diterapkan.

```
root@kci:~# sudo ufw allow 10000/tcp
sudo: unable to resolve host kci: Name or service not known
Rules updated
Rules updated (v6)
```

Gambar 4.23 Menambahkan aturan Firewall

```
root@kci:~# sudo ufw reload
```

Gambar 4.24 Me-reload aturan Firewall

Pada gambar 4.24 *Sudo ufw reload* digunakan untuk me-refresh konfigurasi firewall yang baru saja ditambahkan sebelumnya.

4.3.2 Persiapan Lingkungan Client

Dalam rangka menciptakan skenario pengujian yang komprehensif, penyediaan lingkungan client menjadi krusial untuk simulasi akses pengguna dan observasi

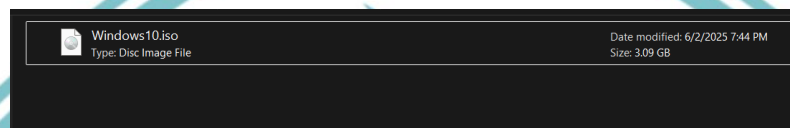
Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

interaksi dengan *web server*. Tahap ini melibatkan konfigurasi mesin virtual dengan sistem operasi Windows yang akan berperan sebagai client pengujian. Tujuannya adalah untuk merepresentasikan perangkat pengguna akhir yang akan mengakses situs web, serta menjadi titik observasi untuk melihat perubahan perilaku peramban saat berkomunikasi dengan web server yang terenkripsi.

Langkah-langkah persiapan lingkungan client adalah sebagai berikut:

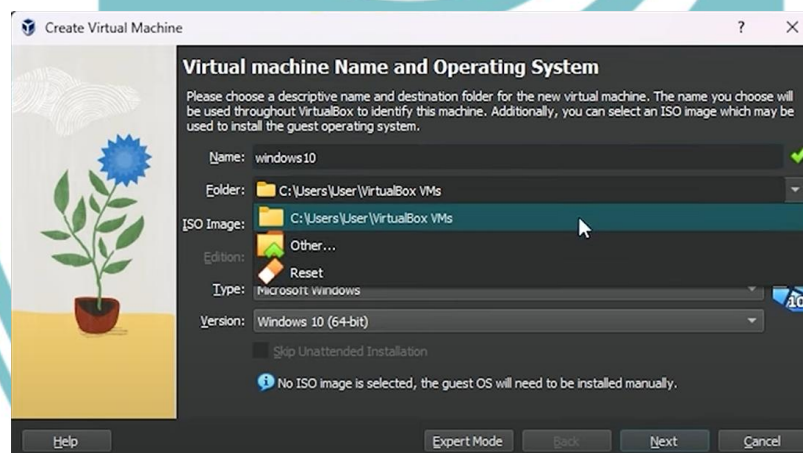
1. File Iso Windows 10



Gambar 4.25 File ISO Windows10

Pada gambar 4.25 ditampilkan file iso Windows 10 yang akan digunakan untuk melakukan instalasi OS di virtual box.

2. *Virtual Machine Name*



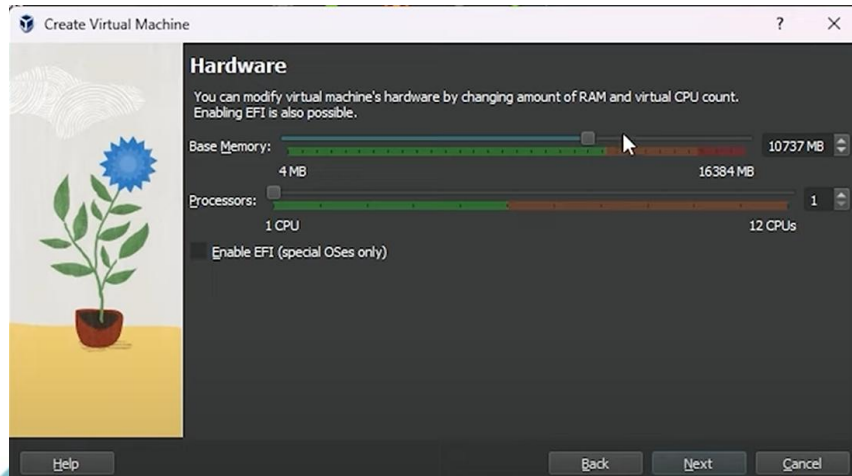
Gambar 4.26 Virtual Machine name Windows10

Pada gambar 4.26 ditampilkan kolom *Virtual Machine*. Pada tahap ini nama pada virtual machine dan pilih ISO windows 10 akan di konfigurasi.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

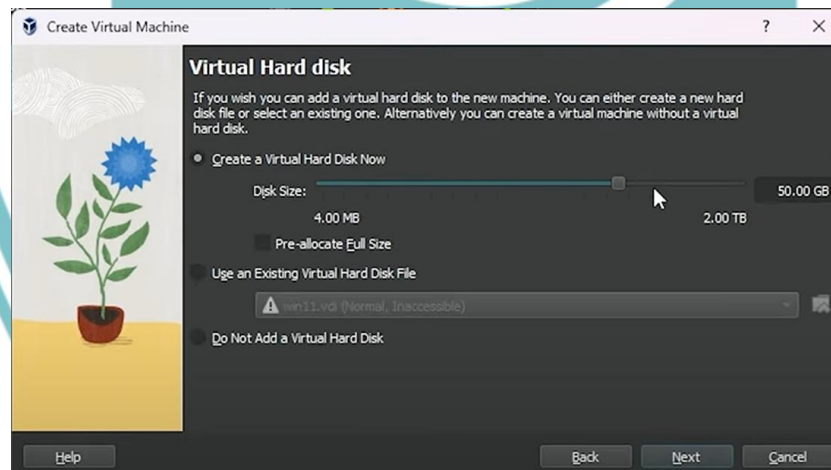
3. Hardware Configuration



Gambar 4.27 Hardware Configuration Windows10

Pada gambar 4.27 ditampilkan *Hardware Configuration*. Pada tahap ini konfigurasi base memory dan prosesor setinggi mungkin agar windows 10 dapat berjalan dengan baik di virtual box.

4. Virtual Hard disk



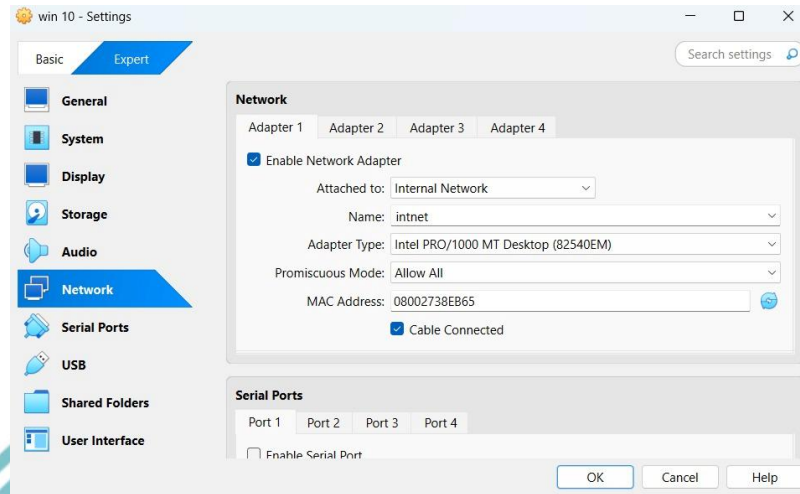
Gambar 4.27 Hard disk Configuration Windows10

Pada gambar 4.27 ditampilkan *hard disk configuration*. Sesuaikan Kebutuhan *Hard disk* windows 10 untuk melakukan penelitian.

5. Network Configuration

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Gambar 4.28 Pengaturan Adaptor Jaringan Internal Virtualbox

Pada gambar 4.28 ditampilkan *Network Configuration*. Konfigurasi ini diperlukan untuk memastikan komunikasi yang terkontrol antara semua VM (*server*, *client*, dan *attacker*), adaptor jaringan mesin virtual Windows dikonfigurasi sebagai Internal Network. Pengaturan ini menciptakan segmen jaringan virtual terisolasi yang hanya dapat diakses oleh VM-VM yang terhubung pada "Internal Network" yang sama (dalam hal ini, dengan nama *intnet*), tanpa terhubung langsung ke jaringan *host* atau internet eksternal. Mode promiskuus diatur ke "*Allow All*" untuk memungkinkan alat analisis jaringan seperti Wireshark atau Network Miner dapat menangkap semua paket yang melewati adaptor, bukan hanya yang ditujukan ke VM itu sendiri.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

6. Windows 10 Setup



Gambar 4.29 Setup Windows10

Pada gambar 4.29 ditampilkan jendela setup windows 10. Saat pertama kali booting windows 10 tampilan seperti ini akan muncul, sesuaikan bahasa, format waktu dan keyboard input lalu pilih Install now seperti pada gambar 4.30.

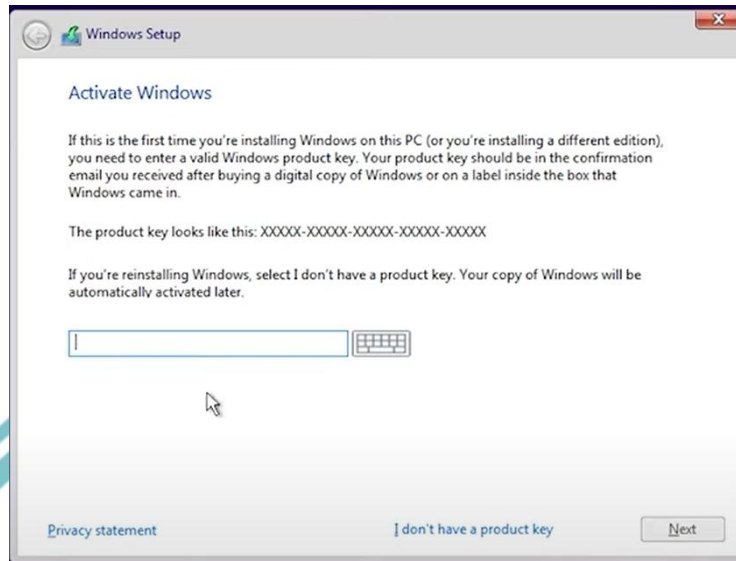


Gambar 4.30 Install now Windows10

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

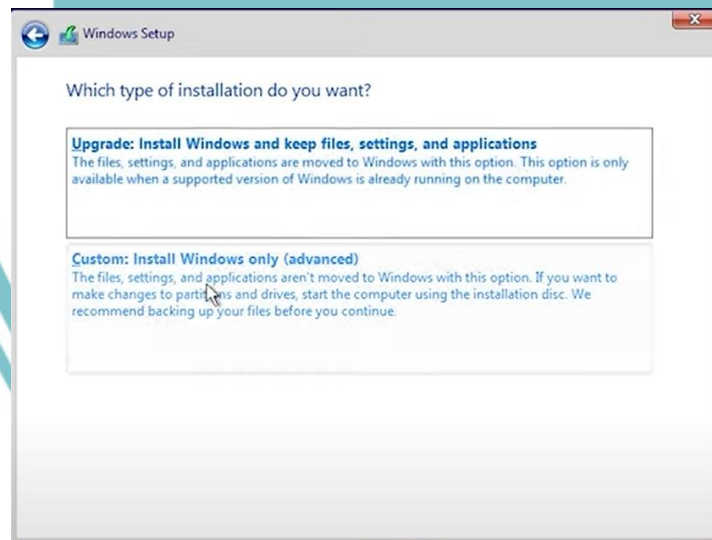
7. Aktivasi Windows



Gambar 4.31 Activation Windows10

Pada gambar 4.31 ditampilkan jendela aktivasi windows. Jika muncul tampilan seperti ini pilih *"I don't have a product key"* untuk melanjutkan instalasi, karena windows 10 dibutuhkan hanya sebagai client pengujian saja.

8. Installation Type

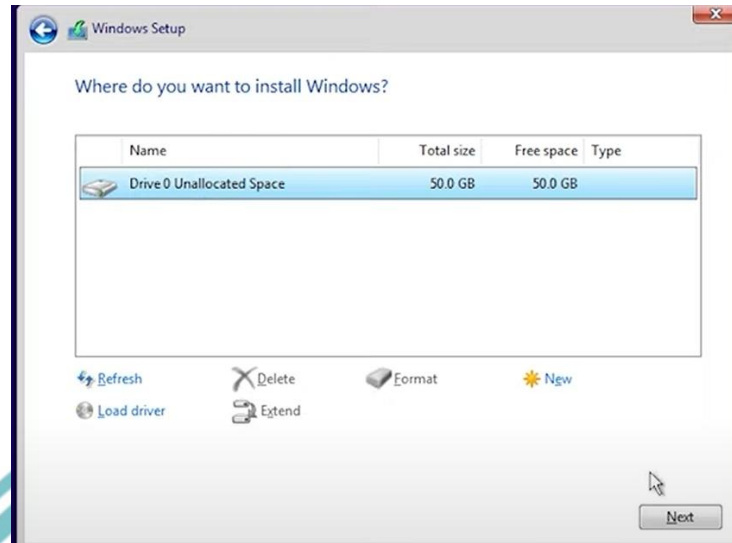


Gambar 4.31 Custom Install Windows10

Pada gambar 4.31 akan diberi pilihan tipe instalasi, pilih *Custom Install* untuk menginstall windows pada *virtual box*, setelah itu windows akan mengarahkan ke proses pemilihan drive seperti pada gambar 4.32.

Hak Cipta :

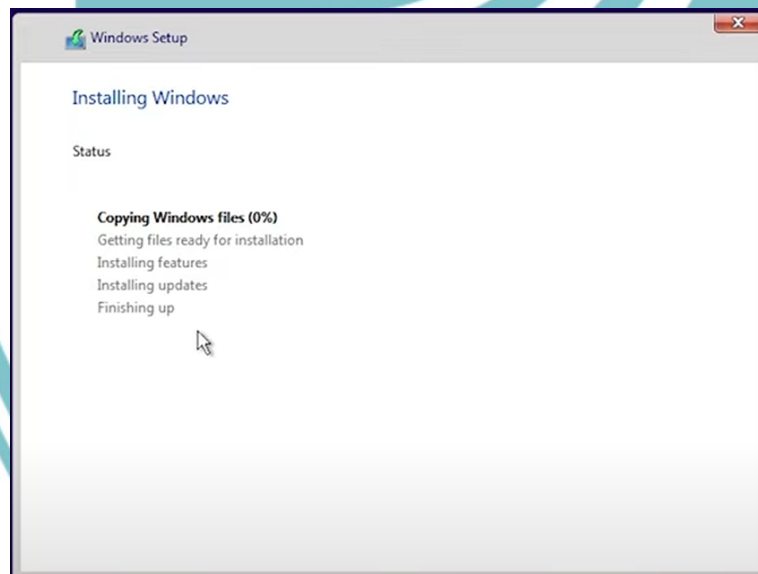
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Gambar 4.32 Unallocated Space Windows10

Pada gambar 4.32 ditampilkan pilihan drive untuk melakukan instalasi windows. Pilih disk yang muncul, lalu klik next.

9. Install Windows

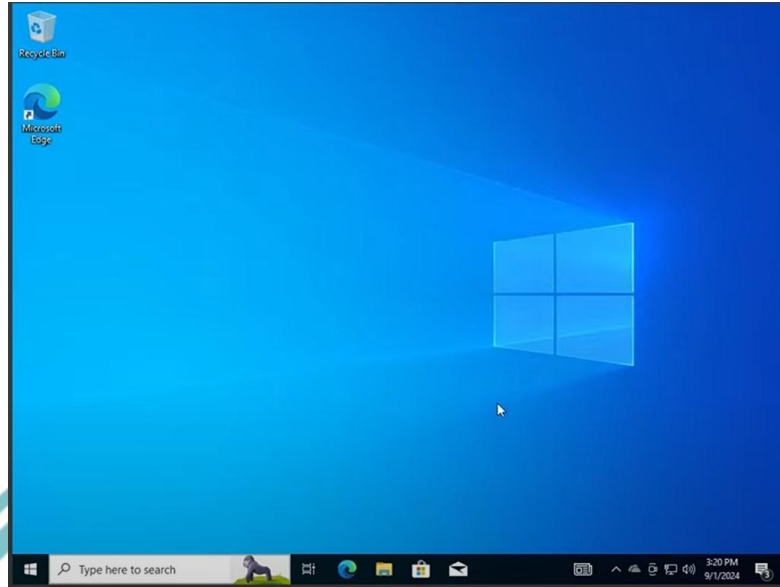


Gambar 4.33 Installing Windows10

Pada gambar 4.33 ditampilkan proses instalasi windows 10. Setelah instalasi selesai virtual box akan *reboot* windows 10 secara otomatis dan windows 10 siap digunakan seperti pada gambar 4.34.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Gambar 4.34 Home Windows10

10. Verifikasi Konfigurasi IP Jaringan Client

Setelah konfigurasi jaringan di Virtual Box, konfigurasi alamat IP pada sistem operasi Windows diverifikasi menggunakan perintah `ipconfig` pada Command Prompt. Observasi menunjukkan bahwa mesin client berhasil mendapatkan alamat IPv4, subnet mask, dan default gateway yang sesuai dengan segmen jaringan internal yang telah dirancang. Contoh konfigurasi IP yang diperoleh adalah alamat IPv4 192.168.57.10, Subnet Mask 255.255.255.0, dan Default Gateway 192.168.57.2. Konfigurasi ini memastikan client dapat berkomunikasi dengan web server virtual dan mesin attacker dalam lingkungan yang sama.

4.3.3 Instalasi dan Akses Webmin

Setelah persiapan dasar server selesai, Webmin diinstal dan diakses untuk manajemen sistem. Prosedur peng-instalasian dan akses webmin antara lain:

1. Instalasi Webmin

Pada gambar 4.35 ditampilkan perintah untuk instalasi webmin. Webmin diinstal menggunakan manajer paket apt setelah *repository* ditambahkan.

```
root@kci:~# sudo apt install webmin
```

Gambar 4.35 Proses Instalasi Webmin

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

2. Verifikasi Status Layanan Webmin

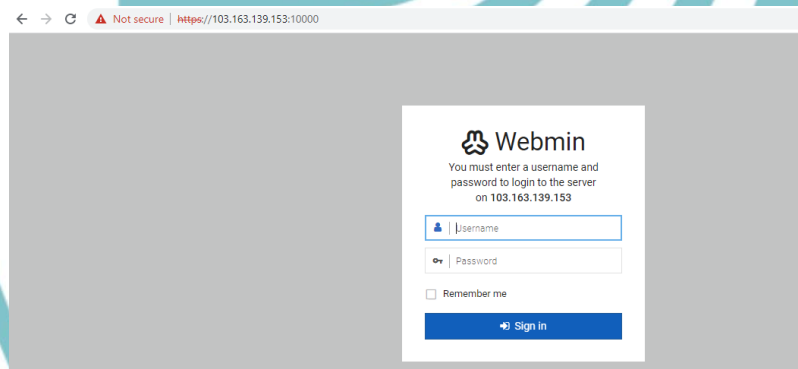
Gambar 4.6 menampilkan status layanan Webmin yang akan diperiksa untuk memastikan bahwa ia berjalan aktif dan siap digunakan.

```
root@kcl:~# systemctl status webmin
● webmin.service - Webmin server daemon
   Loaded: loaded (/lib/systemd/system/webmin.service; enabled; vendor preset: enabled)
   Active: active (running) since Mon 2022-12-12 17:54:37 CET; 2 weeks 2 days ago
     Main PID: 5938 (perl)
        Tasks: 4 (limit: 1074)
       Memory: 55.8M
      CGroup: /system.slice/webmin.service
              └─ 5938 /usr/bin/perl /usr/share/webmin/miniserv.pl --nofork /etc/webmin/miniserv.conf
                 115705 /usr/bin/perl /usr/share/webmin/miniserv.pl --nofork /etc/webmin/miniserv.conf
                 196545 /usr/bin/perl /usr/share/webmin/miniserv.pl --nofork /etc/webmin/miniserv.conf
                 281597 /usr/bin/perl /usr/share/webmin/miniserv.pl --nofork /etc/webmin/miniserv.conf
```

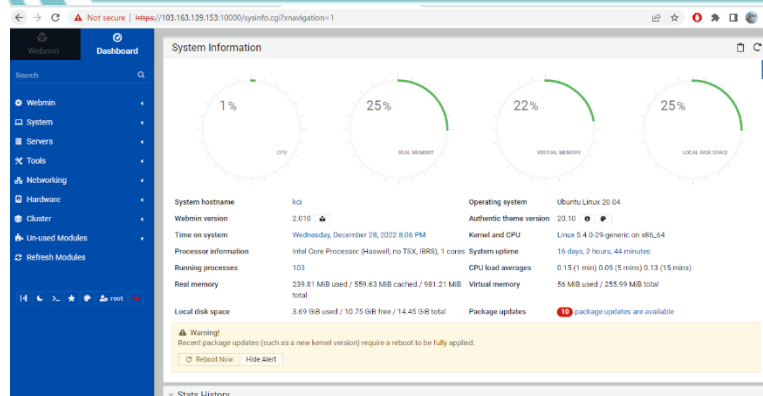
Gambar 4.36 Status Layanan Webmin Setelah Instalasi

3. Akses dan Dashboard Webmin

Gambar 4.37 menampilkan webmin diakses melalui peramban web. Setelah berhasil login, dashboard sistem akan ditampilkan, menyediakan ringkasan informasi server dan platform manajemen yang bisa dilihat pada gambar 4.38.



Gambar 4.37 Tampilan Login Webmin



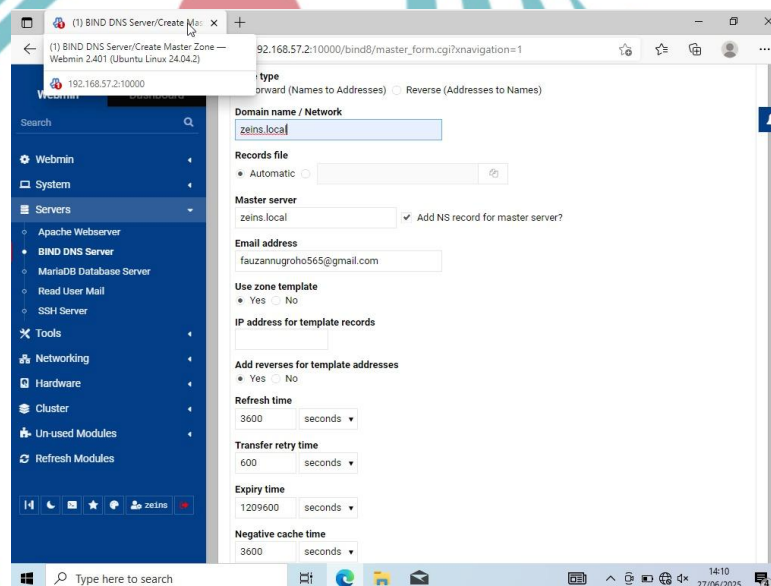
Gambar 4.38 Tampilan Dashboard Webmin

4.3.4 Konfigurasi Prasyarat Web (DNS, Database, Apache Virtual Host)

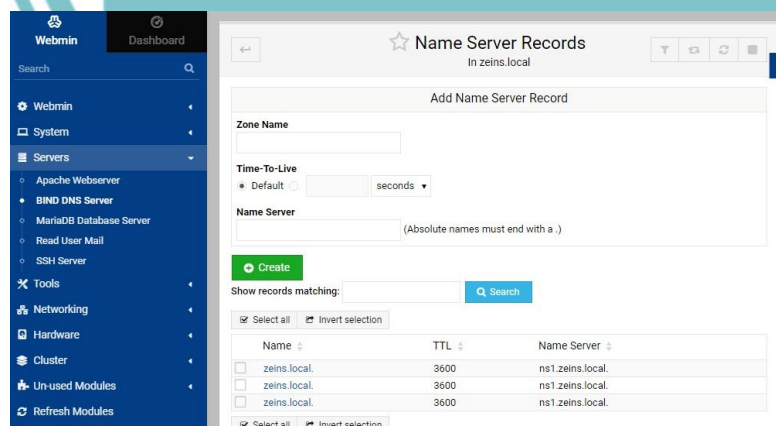
Sebelum mengaktifkan SSL, prasyarat seperti konfigurasi DNS, *database* untuk Webmin, dan *virtual host* Apache harus disiapkan.

1. Konfigurasi DNS (Domain Name System)

Gambar 4.39 menampilkan Pengaturan DNS yang dilakukan untuk memastikan domain name zeins.local mengarah dengan benar ke alamat IP web server. Ini melibatkan pembuatan master zone dan penambahan Address Records (A Records) serta Name Server Records (NS Records) yang dapat dilihat pada gambar 4.40.



Gambar 4.39 Konfigurasi Master Zone



Gambar 4.40 Konfigurasi Name Server

Hak Cipta :

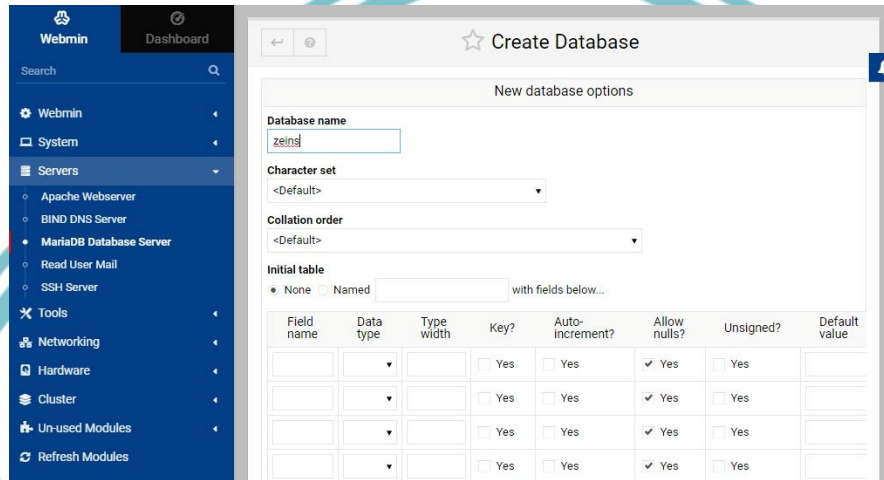
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Hak Cipta :

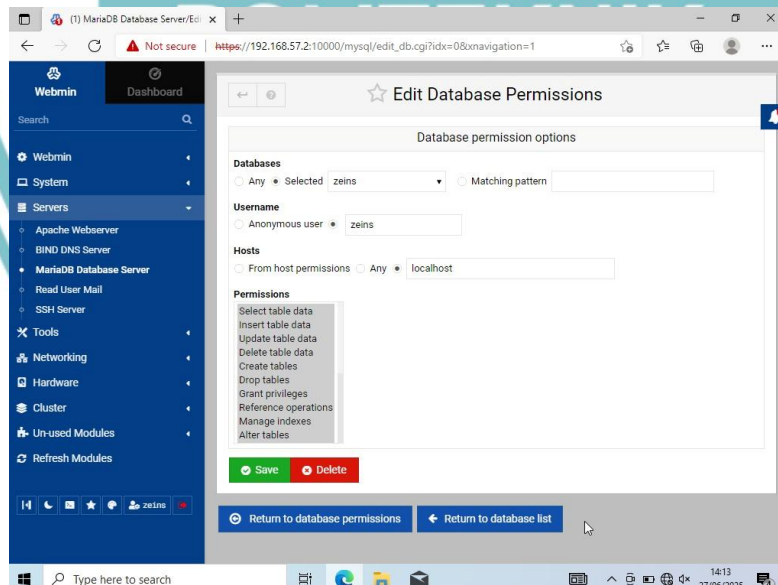
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

2. Pengaturan Database untuk Webmin

Pada gambar 4.41 ditampilkan tampilan untuk pembuatan databse baru. Database baru dibuat misalnya zeins dibuat di MariaDB melalui Webmin, beserta pembuatan user dan penetapan permission yang diperlukan untuk Website webmin seperti pada gambar 4.42 dan 4.43. Pembuatan user FTP juga dilakukan untuk mempermudah transfer file ke server.



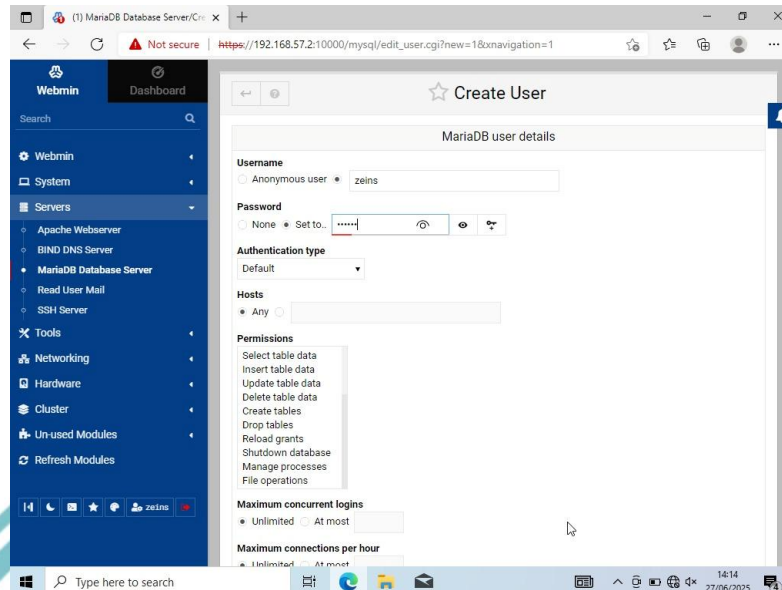
Gambar 4.41 Pembuatan Database



Gambar 4.42 Mengatur Permission Database

Hak Cipta :

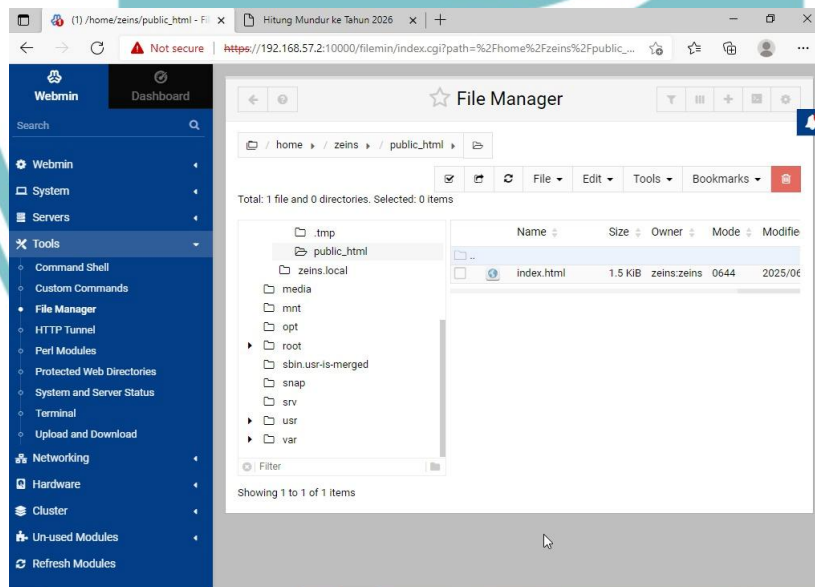
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Gambar 4.43 Membuat User

3. Manajemen File dan Direktori

Pada gambar 4.44 ditampilkan tampilan file manager webmin. Verifikasi struktur direktori public_html menggunakan File Manager Webmin untuk memastikan kesiapan tempat file website.



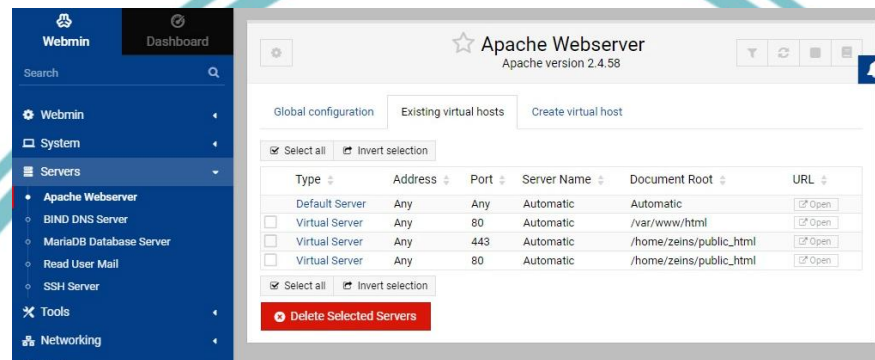
Gambar 4.44 Struktur Direktori Public HTML via File Manager Webmin

Hak Cipta :

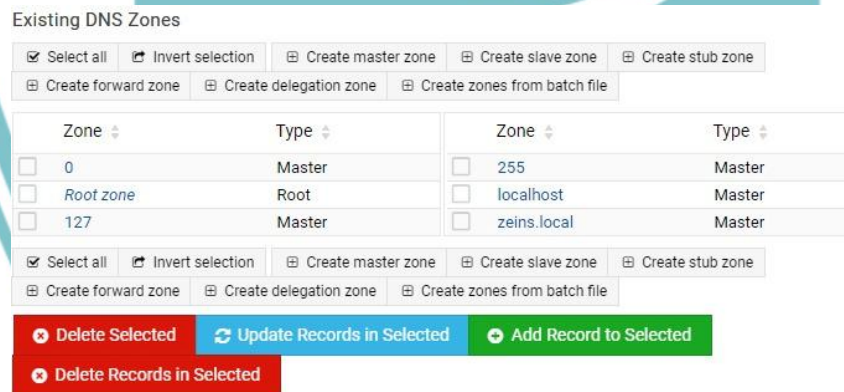
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

4. Pembuatan dan Konfigurasi Virtual Host Apache

Gambar 4.45 menampilkan menu untuk pembuatan *virtual host*. Virtual host Apache dibuat untuk domain zeins.local pada port 80. Ini mengarahkan permintaan ke Document Root yang benar (/home/zeins.local/public_html). Kemudian, daftar virtual host yang sudah diverifikasi, memastikan virtual host untuk port 80 dan port 443 (untuk HTTPS, yang akan dikonfigurasi lebih lanjut) telah terdaftar.



Gambar 4.45 Pembuatan Virtual Server

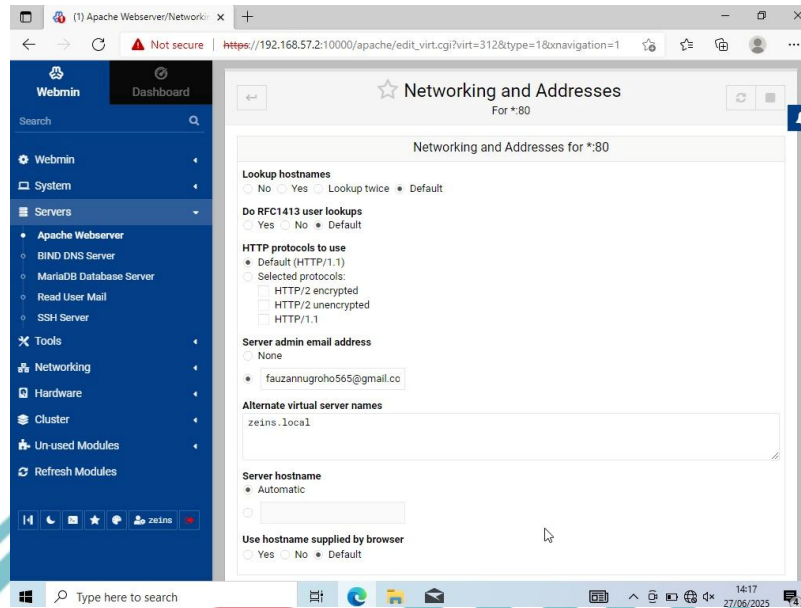


Gambar 4.46 Edit Virtual Host

Gambar 4.46 menampilkan *Existing Dns Zone* yang merupakan konfigurasi penanda dari DNS yang sudah di masukkan saat konfigurasi *virtual host*.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Gambar 4.47 Konfigurasi Network dan Address

Pada gambar 4.47 ditampilkan tampilan *networking and address* yang merupakan konfigurasi jaringan dari virtual host yang sudah dibuat di gambar 4.45.

4.3.5 Implementasi SSL/TLS Menggunakan Sertifikat Self-Signed via OpenSSL dan Webmin

Tahapan ini merupakan realisasi langsung dari perancangan sistem keamanan yang telah dijelaskan sebelumnya, dengan fokus pada aktivasi protokol HTTPS menggunakan sertifikat yang ditandatangani sendiri (*self-signed certificate*) karena lingkungan pengujian bersifat lokal. Dokumentasi ini akan menjelaskan prosedur teknis pembuatan sertifikat SSL/TLS menggunakan OpenSSL dan konfigurasinya pada *Apache Web Server* melalui panel kontrol Webmin. Implementasi ini bertujuan untuk mengaktifkan komunikasi terenkripsi melalui HTTPS di lingkungan pengujian yang terisolasi.

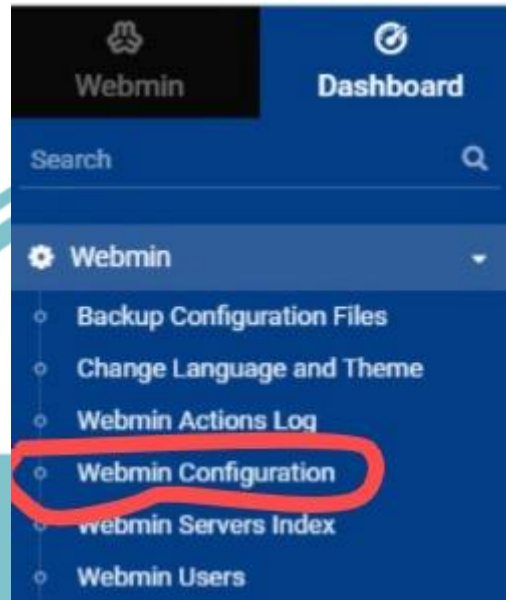
Proses implementasi dilakukan dalam beberapa tahapan kunci, yang meliputi persiapan direktori, pembuatan sertifikat Certificate Authority (CA) lokal, pembuatan sertifikat untuk domain target, konfigurasi SSL di Webmin, hingga verifikasi awal.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

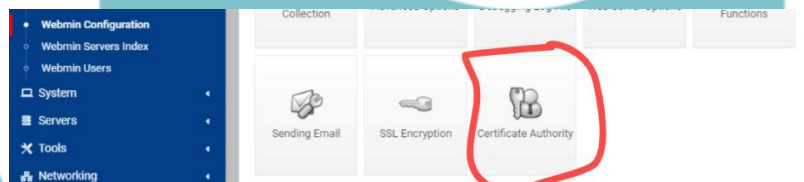
1. Webmin Configuration

Pada gambar 4.48 ditampilkan sidebar dashboard webmin. Buka *sidebar webmin* dan pilih *webmin configuration*



Gambar 4.48 Konfigurasi Webmin

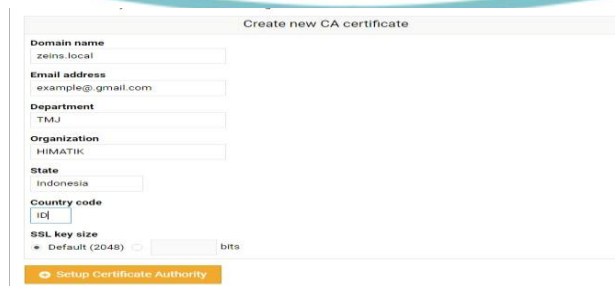
2. Certificate Authority



Gambar 4.49 Certificate Authority

Pada gambar 4.49 ditampilkan menu *Certificate Authority*. Pilih pada bagian *Certificate Authority* untuk membuat CA baru di webmin.

3. Create New CA



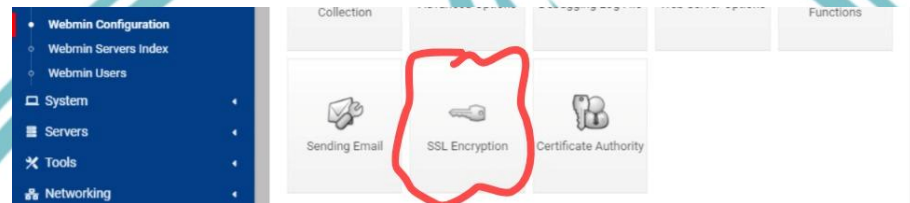
Gambar 4.50 Create New CA

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Pada gambar 4.50 ditampilkan kolom *Create New CA*. Webmin akan menampilkan tampilan *Create new CA certificate* isi keterangan mulai dari domain di isi dengan zeins.local sesuai dengan domain yang akan digunakan, lalu isi email, department dan sebagainya. Setelah semua detail di isi klik “*Setup Certificate Authority*”.

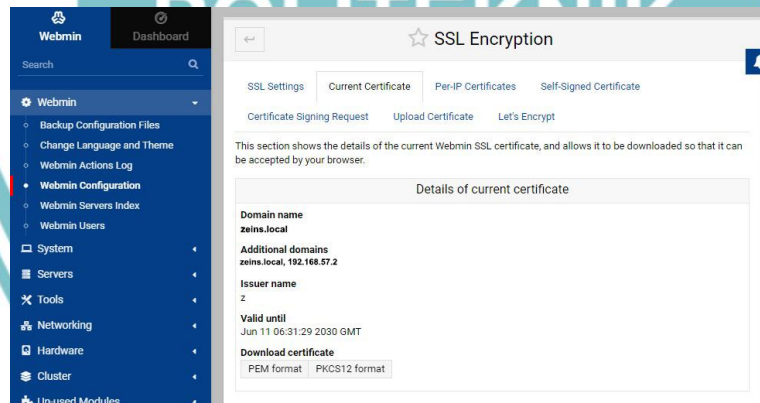
4. SSL Encryption



Gambar 4.51 SSL Encryption

Pada gambar 4.51 ditampilkan menu *SSL Encryption*. Kembali ke bagian *webmin configuration* dan pilih “*SSL Encryption*” untuk mengaktifkan Enkripsi SSL.

5. Current Certificate



Gambar 4.52 Current Certificate

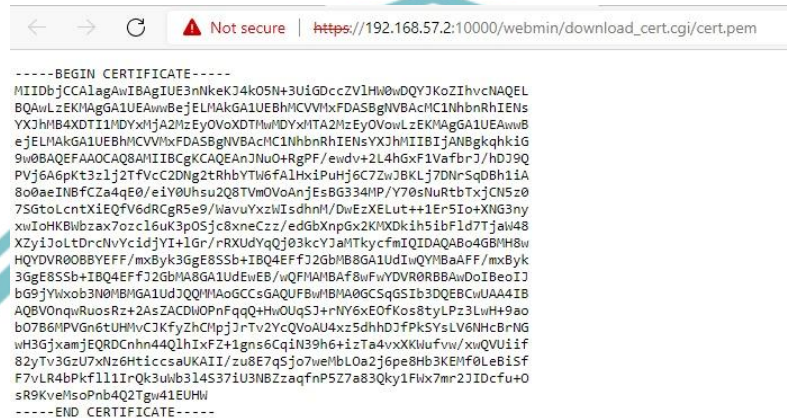
Pada gambar 4.52 ditampilkan menu *SSL Encryption*. Pilih *current certificate* dan isi detail SSL encryption dengan mengetikkan nama domain, dan juga ip address domain pada additional domains, kemudian download certificate menggunakan PEM format dan simpan di client seperti pada gambar 4.53.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



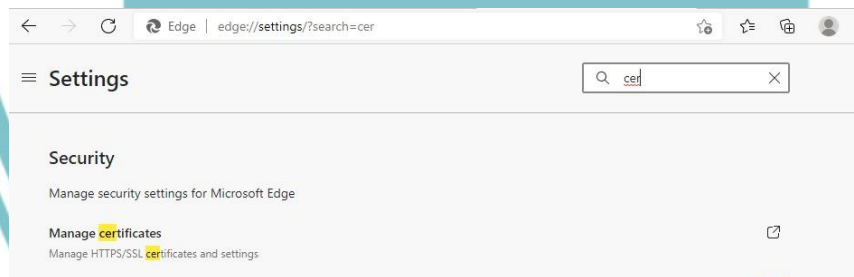
Gambar 4.53 New CA



Gambar 4.54 Certificate Detail

Pada gambar 4.54 ditampilkan isi dari file *certificate* setelah dibuka.

6. Browser Setting

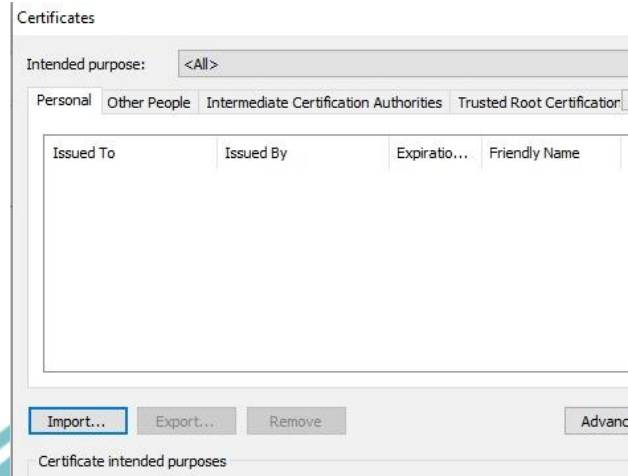


Gambar 4.55 Manage Certificate

Pada gambar 4.55 ditampilkan Pada *browser setting* pilih “*manage certificate*” untuk mengatur sertifikat CA.

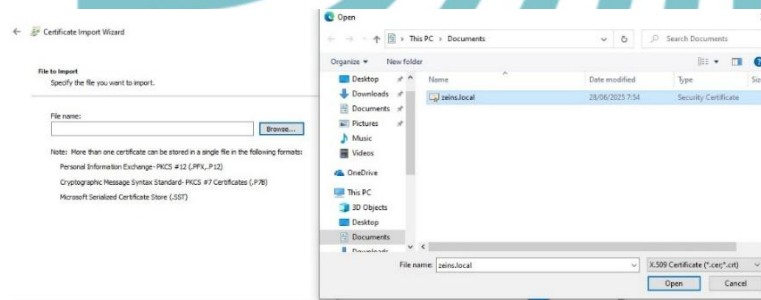
Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Gambar 4.56 Import Certificates CA

Gambar 4.56 merupakan tampilan *import certificates* CA. Setelah muncul tampilan seperti ini pilih *import* dan pilih *browse* seperti pada gambar 4.57.



Gambar 4.57 Browse Certificate CA

7. Verifikasi Awal Konfigurasi SSL/TLS

Pada gambar 4.58 ditampilkan tampilan verifikasi awal SSL. Akses situs web melalui HTTPS (<https://zeins.local:10000> atau <https://zeins.local>) dari peramban. Meskipun koneksi akan menggunakan HTTPS, peramban akan menampilkan peringatan keamanan ("Not secure" / gembok merah) karena sertifikat ditandatangani sendiri dan tidak diakui oleh CA publik.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Gambar 4.58 Verifikasi awal SSL/TLS

Dengan selesainya tahapan implementasi ini, Apache Web Server telah berhasil dikonfigurasi untuk melayani situs web melalui HTTPS dengan sertifikat self-signed, yang dikelola secara efisien melalui Webmin. Sistem kini siap untuk tahapan pengujian guna memvalidasi efektivitas keamanan yang telah diterapkan.

4.4 Pengujian Sistem Keamanan

4.4.1 Deskripsi Pengujian

Pengujian ini tidak semata-mata dilakukan untuk memverifikasi fungsionalitas, melainkan secara spesifik ditujukan untuk memvalidasi keberhasilan konfigurasi Self-signed CA melalui Webmin, serta menganalisis secara mendalam sejauh mana penerapan sertifikat SSL/TLS ini berkontribusi dalam meningkatkan postur keamanan keseluruhan dari situs web yang diuji.

Cakupan pengujian akan meliputi beberapa aspek krusial: pertama, verifikasi fungsionalitas akses HTTPS yang aman; kedua, pengecekan validitas dan detail sertifikat SSL/TLS yang telah diterbitkan; dan ketiga, evaluasi kualitas enkripsi yang diimplementasikan pada koneksi. Pendekatan yang digunakan dalam pengujian ini bersifat komparatif, yaitu membandingkan secara sistematis kondisi web server sebelum dan sesudah penerapan SSL/TLS. Hal ini bertujuan untuk mengidentifikasi perubahan signifikan pada indikator keamanan dan perilaku koneksi. Dengan demikian, hasil pengujian yang diperoleh akan menjadi data

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

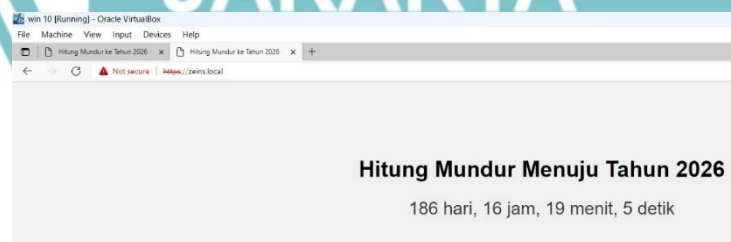
empiris yang krusial untuk menjawab perumusan masalah, khususnya mengenai cara kerja Self-signed CA dalam mengamankan web server dan bentuk perubahannya, sekaligus mencapai tujuan penelitian dalam menganalisis kelebihan serta kekurangan tools ini.

4.4.2 Prosedur Pengujian

Prosedur pengujian dirancang untuk secara sistematis mengevaluasi efektivitas implementasi *Self-signed CA* pada *Apache Web Server* melalui Webmin, serta menganalisis tingkat keamanan yang dicapai. Pengujian ini dilakukan dengan pendekatan komparatif, yaitu membandingkan kondisi keamanan *web server* sebelum dan sesudah penerapan enkripsi SSL/TLS. Seluruh tahapan pengujian dicatat dan didokumentasikan untuk analisis data di sub-bab selanjutnya.

Langkah-langkah pengujian yang dilakukan adalah sebagai berikut:

1. Pengujian Keamanan Dasar (Kondisi HTTP vs. HTTPS):
 - a. Akses Situs Web via HTTP (Sebelum Enkripsi): Buka peramban web dan akses situs web yang di-*hosting* pada *Apache Web Server* menggunakan protokol HTTP (contoh: <http://zeins.local>). Catat indikator keamanan yang ditampilkan oleh peramban, seperti status "Not Secure" atau ikon gembok terbuka pada bilah alamat.



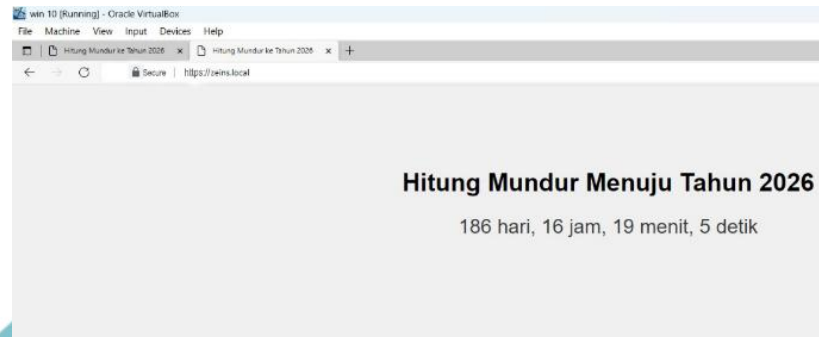
Gambar 4.59 Indikator Koneksi HTTP yang tidak aman

- b. Akses Situs Web via HTTPS (Setelah Enkripsi): Akses situs web melalui peramban web menggunakan protokol HTTPS (contoh: <https://zeins.local>). Verifikasi pengalihan otomatis dari HTTP ke HTTPS.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Amati indikator keamanan pada bilah alamat peramban, seperti ikon gembok terkunci, yang menandakan koneksi aman dan terenkripsi.



Gambar 4.60 Indikator Koneksi HTTPS yang sudah aman

- c. Inspeksi Detail Sertifikat SSL/TLS: Lakukan inspeksi detail sertifikat di peramban untuk memverifikasi penerbit, masa berlaku, dan cakupan *domain* sertifikat SSL/TLS yang telah diinstal.
- d. Analisis Kualitas SSL/TLS dengan Alat Eksternal: Gunakan layanan Qualys SSL Labs SSL Test untuk mendapatkan penilaian komprehensif terhadap konfigurasi SSL/TLS server, termasuk kekuatan *cipher suites* dan dukungan protokol.
- e. Pengecekan Campuran Konten (*Mixed Content*): Periksa konsol *Developer Tools* peramban untuk memastikan tidak ada peringatan *mixed content* yang dapat mengindikasikan pemuatan sumber daya tidak aman di halaman HTTPS.

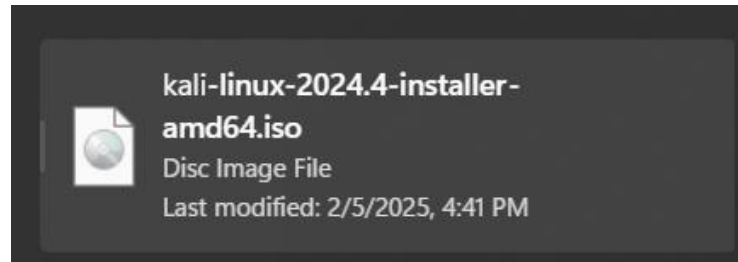
2. Simulasi Serangan untuk Pengujian Ketahanan Sistem

Untuk melakukan Pengujian Ketahanan Sistem penulis memerlukan *attacker side*. Untuk membuat *attacker side* penulis akan menggunakan kali linux sebagai hostnya. Kali Linux dipilih karena Kali Linux adalah distro Linux yang diperuntukan khusus untuk melakukan *Hacking* dan dilengkapi dengan tools tools lengkap yang mendukung untuk melakukan test ketahanan sistem . Untuk menginstall kali Linux langkah langkahnya adalah sebagai berikut:

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

1. File ISO kali linux



Gambar 4.61 File ISO Installer Kali Linux

Pada gambar 4.61 di tampilkan file iso Kali linux yang akan digunakan untuk instalasi OS Kali linux di virtualbox.

2. Kali Linux Installer

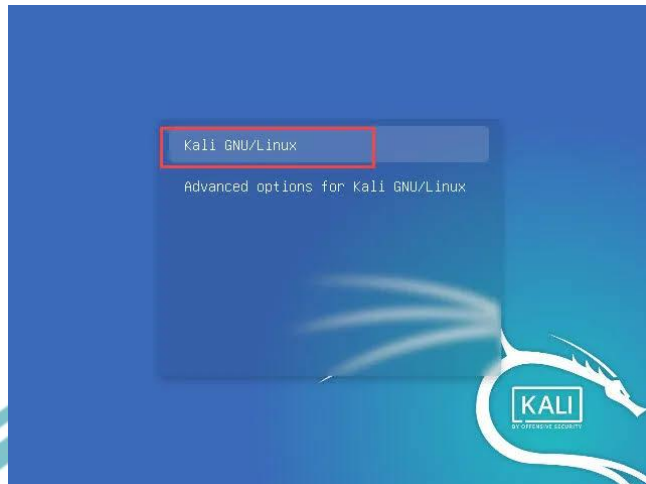


Gambar 4.62 Bios mode Kali Linux

Pada gambar 4.62 ditampilkan tampilan Bios mode kali linux. Bios mode ini merupakan jembatan awal untuk melakukan setup instalasi OS Kali linux. Graphical Instal dipilih agar memudahkan instalasi Kali Linux dengan menggunakan basis grafis. Setelah memilih *graphical install* akan tampil pilihan Kali/GNU linux seperti pada gambar 4.63 dibawah.

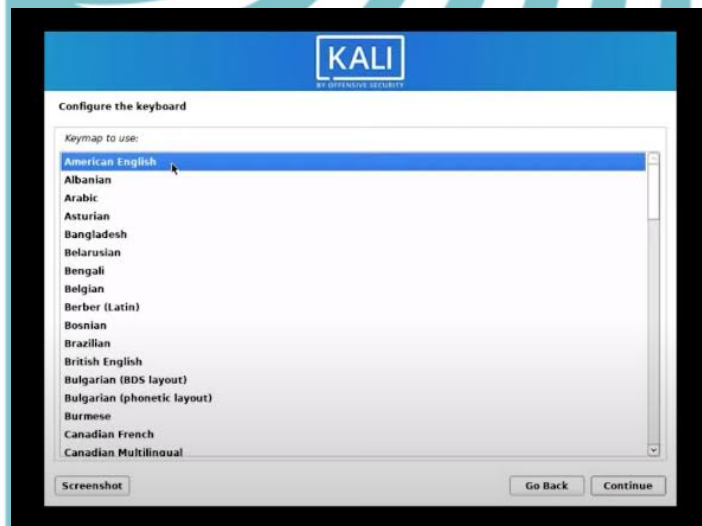
Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Gambar 4.63 Installer Kali GNU/Linux

3. Keyboard Layout



Gambar 4.64 Konfigurasi Keyboard

Pada gambar 4.64 ditampilkan konfigurasi keyboard. Konfigurasi ini digunakan untuk memilih layout keyboard yang akan digunakan untuk sistem.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

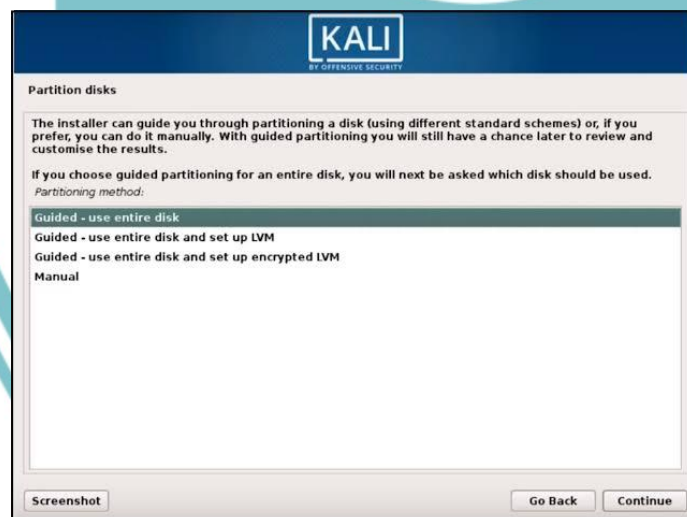
4. Kali Linux Password



Gambar 4.65 Pembuatan Password Kali Linux

Pada gambar 4.65 ditampilkan kolom Password kali linux. Tampilan ini digunakan untuk mengatur password yang akan digunakan dalam sistem kali linux.

5. Partisi Kali Linux



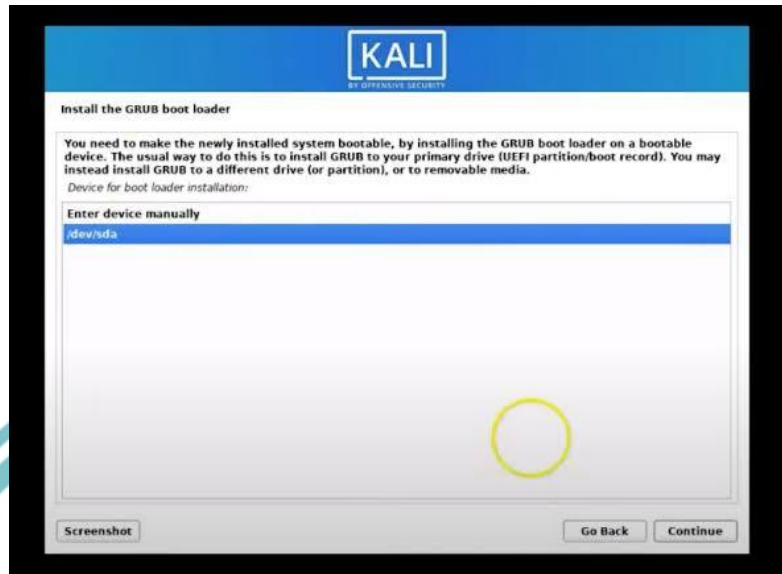
Gambar 4.66 Pemilihan Partisi Instalasi Kali Linux

Pada gambar 4.66 ditampilkan tampilan konfigurasi partisi kali linux. Konfigurasi ini diperlukan untuk menentukan partisi tempat install sistem operasi Kali linux.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

6. GRUB Boot Loader Kali Linux



Gambar 4.67 Pemilihan partisi instalasi GRUB

Pada gambar 4.67 ditampilkan tampilan instalasi GRUB Boot Loader kali linux, yang merupakan tahapan terakhir instalasi kali linux. Tahapan ini digunakan untuk menentukan partisi tempat penginstallan GRUB Kali linux.

Setelah melakukan Penginstallan sistem operasi kali linux sebagai *attacker side* langkah selanjutnya adalah melakukan pengujian ketahanan sistem menggunakan kali linux sebagai penyerang atau attacker. Pengujian ini dilakukan dalam lingkungan lab virtual yang terisolasi dan terkontrol, hanya untuk tujuan edukasi dan uji penetrasi etis. Tujuannya adalah untuk menguji kerentanan web server terhadap jenis serangan tertentu dan mengevaluasi efektivitas HTTPS sebagai mitigasi.

4.4.3 Data Hasil Pengujian

Data hasil pengujian disajikan secara sistematis berdasarkan prosedur yang telah ditetapkan dalam sub-bab 4.4.2. Seluruh observasi dan bukti visual ini merupakan data empiris yang akan menjadi dasar analisis di bagian selanjutnya. Pengujian dilakukan dengan membandingkan kondisi keamanan situs web sebelum dan

sesudah implementasi sertifikat SSL/TLS dari Self-signed CA melalui Webmin, serta mencakup hasil simulasi serangan Man-in-the-Middle (MITM).

1. Hasil Pengujian Keamanan Dasar (Kondisi HTTP vs. HTTPS)

Pengujian ini merekam perubahan indikator keamanan dan perilaku koneksi situs web setelah penerapan sertifikat SSL/TLS.

a. Kondisi Sebelum Implementasi SSL/TLS (HTTP):

Dapat dilihat website yang masih HTTP jika dianalisis menggunakan *Network Miner* maka port yang dihasilkan adalah port 80 (HTTP) yang menandakan bahwa website belum secure.



Gambar 4.68 Website HTTP

b. Kondisi Sesudah Implementasi SSL/TLS (HTTPS) dengan Sertifikat Self-Signed:

Kondisi website setelah SSL, jika dilihat dengan aplikasi *Network Miner* maka akan didapatkan port 443 (SSL) yang menandakan bahwa website sudah secure dengan self signed.



Gambar 4.69 Website HTTPS

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Hak Cipta :

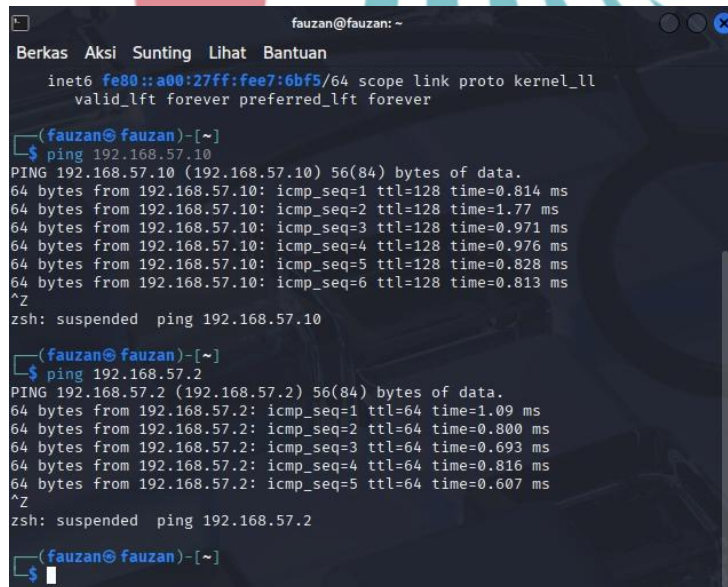
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

2. Hasil Simulasi Serangan Man-in-the-Middle (MITM)

Simulasi serangan MITM dilakukan untuk menguji respons dan ketahanan *web server* terhadap upaya penyadapan data dalam lingkungan lab virtual.

1. Verifikasi Konektivitas Awal Lingkungan Serangan:

Sebelum memulai serangan, konektivitas dasar antara mesin virtual client, server, dan attacker diverifikasi menggunakan perintah ping. Hasil ping menunjukkan bahwa ketiga mesin dapat berkomunikasi satu sama lain di dalam jaringan internal yang terisolasi, memastikan fondasi jaringan untuk simulasi telah siap.



```

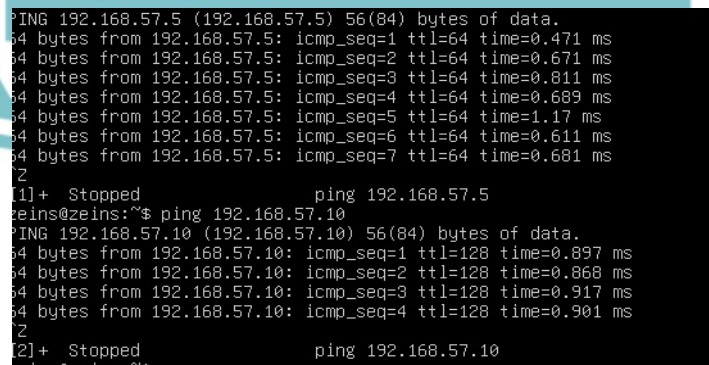
fauzan@fauzan: ~
Berkas Aksi Sunting Lihat Bantuan
inet6 fe80::a00:27ff:fee7:6bf5/64 scope link proto kernel_ll
valid_lft forever preferred_lft forever

(fauzan@fauzan)-[~]
$ ping 192.168.57.10
PING 192.168.57.10 (192.168.57.10) 56(84) bytes of data.
64 bytes from 192.168.57.10: icmp_seq=1 ttl=128 time=0.814 ms
64 bytes from 192.168.57.10: icmp_seq=2 ttl=128 time=1.77 ms
64 bytes from 192.168.57.10: icmp_seq=3 ttl=128 time=0.971 ms
64 bytes from 192.168.57.10: icmp_seq=4 ttl=128 time=0.976 ms
64 bytes from 192.168.57.10: icmp_seq=5 ttl=128 time=0.828 ms
64 bytes from 192.168.57.10: icmp_seq=6 ttl=128 time=0.813 ms
^Z
zsh: suspended ping 192.168.57.10

(fauzan@fauzan)-[~]
$ ping 192.168.57.2
PING 192.168.57.2 (192.168.57.2) 56(84) bytes of data.
64 bytes from 192.168.57.2: icmp_seq=1 ttl=64 time=1.09 ms
64 bytes from 192.168.57.2: icmp_seq=2 ttl=64 time=0.800 ms
64 bytes from 192.168.57.2: icmp_seq=3 ttl=64 time=0.693 ms
64 bytes from 192.168.57.2: icmp_seq=4 ttl=64 time=0.816 ms
64 bytes from 192.168.57.2: icmp_seq=5 ttl=64 time=0.607 ms
^Z
zsh: suspended ping 192.168.57.2

(fauzan@fauzan)-[~]
$
  
```

Gambar 4.70 Hasil Ping dari Mesin Attacker ke Client dan Server.



```

PING 192.168.57.5 (192.168.57.5) 56(84) bytes of data.
64 bytes from 192.168.57.5: icmp_seq=1 ttl=64 time=0.471 ms
64 bytes from 192.168.57.5: icmp_seq=2 ttl=64 time=0.671 ms
64 bytes from 192.168.57.5: icmp_seq=3 ttl=64 time=0.811 ms
64 bytes from 192.168.57.5: icmp_seq=4 ttl=64 time=0.689 ms
64 bytes from 192.168.57.5: icmp_seq=5 ttl=64 time=1.17 ms
64 bytes from 192.168.57.5: icmp_seq=6 ttl=64 time=0.611 ms
64 bytes from 192.168.57.5: icmp_seq=7 ttl=64 time=0.681 ms
^Z
[1]+  Stopped                  ping 192.168.57.5
zeins@zeins:~$ ping 192.168.57.10
PING 192.168.57.10 (192.168.57.10) 56(84) bytes of data.
64 bytes from 192.168.57.10: icmp_seq=1 ttl=128 time=0.897 ms
64 bytes from 192.168.57.10: icmp_seq=2 ttl=128 time=0.868 ms
64 bytes from 192.168.57.10: icmp_seq=3 ttl=128 time=0.917 ms
64 bytes from 192.168.57.10: icmp_seq=4 ttl=128 time=0.901 ms
^Z
[2]+  Stopped                  ping 192.168.57.10
zeins@zeins:~$
  
```

Gambar 4.71 Hasil Ping dari Mesin Server ke Attacker dan Client.

Hak Cipta :

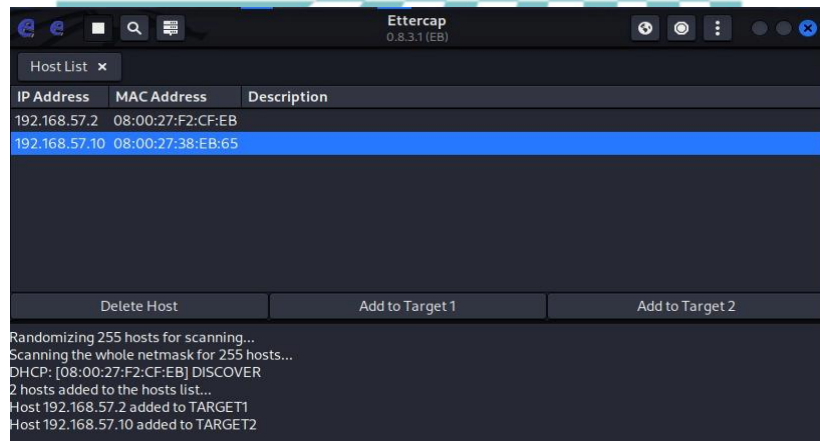
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

2. Hasil Observasi Lalu Lintas MITM dengan Ettercap, Wireshark, dan NetworkMiner:

Dalam simulasi serangan MITM menggunakan Ettercap dan observasi dengan Wireshark/NetworkMiner, perbedaan signifikan terlihat pada visibilitas data tergantung pada protokol yang digunakan.

1. Kondisi HTTP:

Lalu lintas komunikasi yang terjadi saat situs web diakses melalui HTTP dapat disadap dan dianalisis oleh penyerang dalam bentuk *plaintext*. Informasi seperti kredensial (*username* dan *password*) dan konten halaman web menjadi rentan dan dapat dibaca langsung oleh alat seperti Ettercap, Wireshark, atau NetworkMiner.



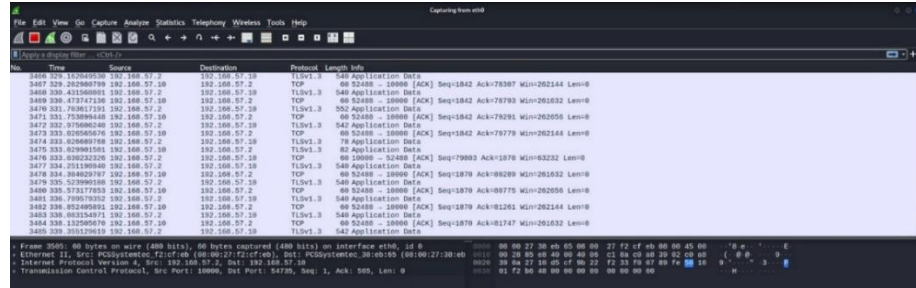
Gambar 4.72 Hasil Penyadapan Data HTTP oleh Ettercap.

2. Kondisi HTTPS:

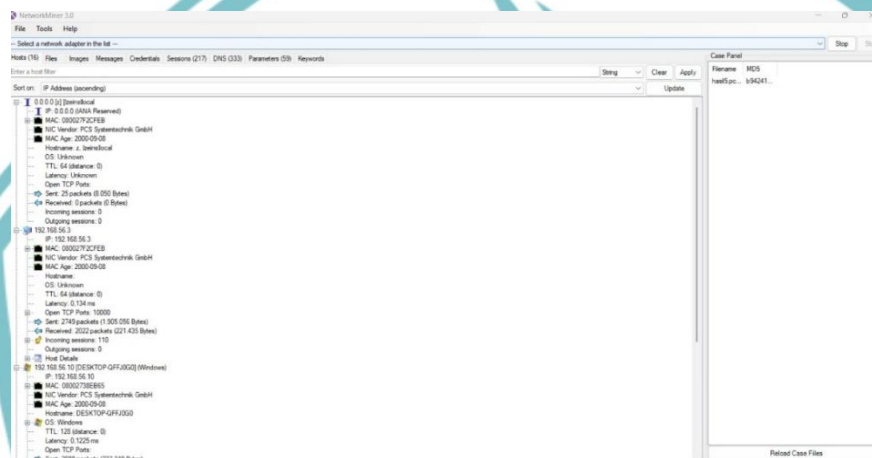
Setelah situs web dienkripsi dengan Self-signed CA, lalu lintas komunikasi yang melewati jalur MITM tetap terenkripsi menggunakan protokol SSL/TLS atau TLSv1.3. Data yang disadap oleh Ettercap, Wireshark, atau NetworkMiner hanya menampilkan *ciphertext* yang tidak dapat diinterpretasikan secara langsung. Kredensial tidak terekam dalam bentuk *plaintext*. Informasi metadata terbatas seperti Server Name Indication (SNI) dan alamat IP tujuan mungkin masih terlihat selama TLS Handshake, namun isi komunikasi inti tetap aman dan rahasia.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Gambar 4.73 Tampilan Paket Terenkripsi TLSv1.3 di Wireshark.



Gambar 4.74 Analisis Host di NetworkMiner dengan Lalu Lintas Terenkripsi.

4.4.4 Analisis Data dan Evaluasi Pengujian

Berdasarkan data hasil pengujian yang telah disajikan pada sub-bab 4.4.3, telah dilakukan analisis mendalam untuk mengevaluasi efektivitas implementasi SSL/TLS menggunakan sertifikat self-signed pada Apache Web Server melalui Webmin dalam meningkatkan keamanan situs web. Evaluasi ini juga bertujuan untuk menjawab permasalahan masalah yang telah ditetapkan.

1. Analisis Perubahan Protokol dan Keamanan Koneksi:

Perbandingan kondisi web server sebelum dan sesudah implementasi SSL/TLS menunjukkan transformasi fundamental dalam cara komunikasi terjadi. Sebelum penerapan sertifikat SSL/TLS, akses situs web melalui protokol HTTP pada zeins.local secara jelas ditandai oleh peramban sebagai koneksi "Not Secure". Hal ini mengindikasikan bahwa data yang



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

ditransmisikan antara client dan server tidak terenkripsi, sehingga rentan terhadap penyadapan dan manipulasi oleh pihak tidak berwenang.

Sebaliknya, setelah implementasi sertifikat self-signed, situs web berhasil diakses melalui HTTPS. Namun, karena sertifikat ini tidak diterbitkan oleh Certificate Authority (CA) publik yang secara default dipercaya oleh peramban, ikon keamanan pada bilah alamat peramban tetap menampilkan indikator "Not secure" (misalnya gembok merah) meskipun protokolnya sudah HTTPS. Terlepas dari peringatan visual ini, perubahan protokol dari HTTP ke HTTPS membuktikan bahwa komunikasi data kini ditransmisikan melalui jalur terenkripsi (SSL/TLS), yang secara signifikan meningkatkan kerahasiaan dan integritas informasi yang dipertukarkan. Efektivitas enkripsi pada lapisan transportasi ini selaras dengan tujuan untuk mengamankan komunikasi web.

2. Evaluasi Validitas dan Kualitas Sertifikat SSL/TLS (Self-Signed):

Pemeriksaan detail sertifikat memverifikasi bahwa sertifikat yang digunakan diterbitkan oleh "Self-signed CA (E6)" sebagai issuer yang merupakan bagian dari rantai kepercayaan yang dibuat secara lokal. Sertifikat ini mencakup domain utama zeins.local dan subdomain terkait (mail.zeins.local, www.zeins.local), serta memiliki masa berlaku 90 hari, sesuai karakteristik Self-signed CA sebagai intermediate certificate dalam rantai kepercayaan lokal Anda. Meskipun teridentifikasi adanya validation error pada status revokasi, dan sertifikat secara default tidak dipercaya oleh peramban publik, fungsionalitas dasar enkripsi tetap terjaga dalam lingkungan terkontrol.

Kualitas konfigurasi SSL/TLS diverifikasi melalui inspeksi detail di peramban dan data sertifikat itu sendiri. Implementasi ini berhasil menyediakan dukungan untuk protokol TLS versi terbaru (TLS 1.3 dan TLS 1.2), serta penggunaan cipher suites yang kuat dan mendukung forward secrecy. Simulasi handshake juga menunjukkan kompatibilitas yang luas dengan berbagai client modern maupun legacy, dan tidak terdeteksinya kerentanan umum seperti Heartbleed, POODLE, atau DROWN. Hal ini menegaskan bahwa implementasi SSL/TLS self-signed, meskipun



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

memerlukan impor CA lokal ke trust store peramban agar dipercaya sepenuhnya, secara teknis mampu menyediakan enkripsi yang fungsional dan memenuhi standar keamanan yang direkomendasikan untuk lapisan transportasi dalam lingkungan lokal.

3. Analisis Ketahanan Terhadap Simulasi Serangan Man-in-the-Middle (MITM):

Pengujian terhadap simulasi serangan MITM menggunakan Ettercap pada lingkungan lab virtual menunjukkan perbedaan signifikan antara komunikasi HTTP dan HTTPS. Verifikasi konektivitas awal antar mesin virtual memastikan lingkungan simulasi berjalan dengan baik. Dalam kondisi HTTP, lalu lintas data yang melewati titik MITM dapat disadap dan diinterpretasikan dalam bentuk plaintext oleh penyerang, yang berpotensi menyebabkan pencurian kredensial atau informasi sensitif lainnya.

Sebaliknya, setelah implementasi SSL/TLS self-signed dan aktivasi HTTPS, lalu lintas data yang melewati jalur MITM tetap terenkripsi. Data yang disadap oleh Ettercap, Wireshark, atau NetworkMiner hanya menampilkan ciphertext yang tidak dapat dibaca atau diinterpretasikan oleh penyerang. Kredensial tidak terekam dalam bentuk plaintext. Ini menegaskan bahwa enkripsi SSL/TLS secara efektif memitigasi risiko penyadapan dan manipulasi data dalam serangan MITM, membuktikan bahwa implementasi SSL/TLS adalah pertahanan yang kuat terhadap ancaman tersebut, sesuai dengan tujuan pengamanan situs web.

4. Evaluasi Komprehensif dan Keterkaitan dengan Perumusan Masalah:

Secara keseluruhan, analisis data pengujian secara komprehensif membuktikan bahwa implementasi sistem keamanan pada Apache Web Server menggunakan sertifikat self-signed via Webmin berhasil secara signifikan meningkatkan postur keamanan situs web dalam lingkungan lokal.

- a. Mengenai cara kerja SSL/TLS dalam mengamankan web server, hasil pengujian menunjukkan bahwa implementasi ini efektif mengubah protokol HTTP menjadi HTTPS, menyediakan enkripsi end-to-end, dan mencegah penyadapan data oleh serangan MITM, terlepas dari status kepercayaan sertifikat oleh peramban.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

- b. Analisis keamanan dari implementasi SSL/TLS pada Webmin menunjukkan bahwa kombinasi ini mempermudah proses administrasi SSL/TLS secara otomatis dan menghasilkan konfigurasi enkripsi yang fungsional, meskipun status "Not secure" di peramban merupakan konsekuensi dari sifat self-signed sertifikat yang tidak diakui CA publik.
- c. Bentuk perubahan pada web server sesudah dan sebelum menggunakan SSL/TLS terlihat jelas dari transisi status "Not Secure" menjadi HTTPS (dengan peringatan keamanan), serta kemampuan enkripsi yang terukur dari pengujian langsung.
- d. Kelebihan utama dari pendekatan ini adalah kemampuan untuk mengaktifkan enkripsi HTTPS di lingkungan lokal yang terisolasi tanpa bergantung pada Certificate Authority publik. Perubahan ini secara langsung mengatasi kerentanan yang ada pada protokol HTTP dan mendukung tujuan penelitian untuk menganalisis efektivitas implementasi SSL/TLS dalam mewujudkan lingkungan web yang lebih aman.

**POLITEKNIK
NEGERI
JAKARTA**

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

BAB V PENUTUP

5.1 Simpulan

Berdasarkan analisis dan pengujian yang telah dilakukan, dapat ditarik beberapa simpulan yang secara langsung menjawab tujuan penelitian:

1. Implementasi Sertifikat Self-Signed CA: Implementasi sertifikat self-signed CA pada Apache Web Server berhasil dilakukan dengan efektif di lingkungan lokal menggunakan kombinasi OpenSSL untuk pembuatan sertifikat dan Webmin sebagai antarmuka manajemen. Proses ini, mulai dari pembuatan Certificate Authority (CA) lokal hingga konfigurasi virtual host di Apache, dapat dikelola dengan lebih sederhana melalui antarmuka grafis Webmin, yang membuktikan kelayakan metode ini untuk lingkungan pengembangan.
2. Analisis Perbandingan Keamanan (HTTP vs. HTTPS): Terdapat peningkatan keamanan yang fundamental setelah implementasi sertifikat self-signed. Sebelum implementasi, koneksi melalui HTTP ditandai "Not Secure" oleh peramban dan rentan terhadap penyadapan plaintext. Setelah implementasi, protokol komunikasi berhasil diubah menjadi HTTPS. Meskipun peramban menampilkan peringatan keamanan karena sifat self-signed dari sertifikat, analisis lalu lintas jaringan membuktikan bahwa seluruh data telah ditransmisikan melalui jalur terenkripsi (SSL/TLS), sehingga kerahasiaan dan integritasnya terjaga.
3. Efektivitas Mitigasi Serangan Man-in-the-Middle (MITM): Implementasi SSL/TLS menggunakan sertifikat self-signed terbukti sangat efektif dalam memitigasi simulasi serangan Man-in-the-Middle di lingkungan pengujian yang terisolasi. Pada koneksi HTTPS, alat seperti Ettercap dan Wireshark tidak mampu menangkap kredensial atau data sensitif dalam bentuk plaintext. Data yang disadap hanya berupa ciphertext yang tidak dapat diinterpretasikan, yang menegaskan bahwa enkripsi pada lapisan transportasi berhasil melindungi komunikasi dari ancaman penyadapan.
4. Kelebihan dan Keterbatasan Teknis: Kelebihan utama dari metode ini adalah kemampuannya untuk mengaktifkan enkripsi HTTPS secara cepat

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

dan tanpa biaya di lingkungan pengembangan lokal yang terisolasi, tanpa memerlukan validasi domain publik. Namun, keterbatasan teknis yang paling signifikan adalah sertifikat self-signed tidak dipercaya secara otomatis oleh peramban publik, sehingga memunculkan peringatan keamanan yang mengharuskan impor CA lokal secara manual ke dalam trust store klien untuk pengujian yang mulus.

5.2 Milestone Penelitian

Selama pelaksanaan penelitian ini, beberapa tonggak pencapaian (milestone) teknis berhasil diraih, yang menjadi fondasi bagi kesimpulan akhir:

1. Penciptaan Lingkungan Laboratorium Virtual yang Fungsional: Berhasil membangun infrastruktur jaringan virtual yang terisolasi menggunakan VirtualBox, yang terdiri dari tiga mesin virtual dengan peran spesifik: Ubuntu sebagai Server, Windows sebagai Client, dan Kali Linux sebagai Attacker. Konfigurasi alamat IP dan jaringan internal yang tepat memastikan semua komponen dapat berkomunikasi sesuai skenario pengujian.
2. Implementasi Penuh Self-Signed Certificate via Webmin: Berhasil mengonfigurasi Apache Web Server untuk menggunakan protokol HTTPS dengan mengimplementasikan sertifikat yang dibuat dan ditandatangani sendiri (self-signed certificate). Proses ini berhasil dikelola seluruhnya melalui antarmuka Webmin, menunjukkan efisiensi Webmin dalam menyederhanakan tugas administrasi SSL/TLS yang kompleks.
3. Demonstrasi Keberhasilan Enkripsi Data: Berhasil membuktikan secara empiris bahwa lalu lintas data pada koneksi HTTPS terenkripsi. Melalui analisis paket menggunakan Wireshark, ditunjukkan bahwa data yang sebelumnya berupa plaintext pada koneksi HTTP kini berubah menjadi ciphertext yang tidak dapat dibaca, memvalidasi fungsi utama SSL/TLS.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

4. Validasi Ketahanan Terhadap Serangan MITM: Berhasil melakukan simulasi serangan Man-in-the-Middle (MITM) dan membuktikan bahwa koneksi HTTPS yang diamankan dengan sertifikat self-signed mampu melindungi data dari penyadapan. Kegagalan dalam menangkap kredensial pada sesi HTTPS menjadi bukti nyata efektivitas mitigasi dari implementasi ini.

5.3 Saran

Berdasarkan simpulan dan hasil penelitian yang telah diperoleh, berikut adalah beberapa saran untuk pengembangan lebih lanjut dan penerapan praktis:

1. Studi Komparatif Berkelanjutan: Untuk analisis yang lebih komprehensif, disarankan untuk melakukan studi komparatif dengan Certificate Authority (CA) lain atau metode pengamanan web server lainnya misalnya Nginx dengan Self-signed CA, atau implementasi SSL/TLS manual dari sisi performa dan kemudahan manajemen dalam skala yang lebih besar. Ini dapat memberikan wawasan lebih lanjut mengenai efisiensi dan skalabilitas solusi yang ada.
2. Pengujian Keamanan yang Lebih Dalam: Melakukan pengujian penetrasi (pentest) yang lebih mendalam dan pemindaian kerentanan secara berkala pada web server dan aplikasi web adalah penting. Hal ini untuk mengidentifikasi celah keamanan yang mungkin tidak terdeteksi oleh pengujian standar, serta memastikan konfigurasi yang aman secara berkelanjutan.
3. Pemantauan Log Keamanan: Implementasi sistem pemantauan log keamanan yang proaktif pada Apache dan sistem Ubuntu dapat membantu mendeteksi anomali atau upaya serangan secara real-time, memungkinkan respons cepat dari administrator. Webmin dapat digunakan untuk mempermudah akses dan analisis log.
4. Edukasi dan Kesadaran Pengguna: Penting untuk terus meningkatkan kesadaran pengguna dan administrator mengenai praktik keamanan siber terbaik, termasuk pentingnya menggunakan HTTPS, memperbarui perangkat lunak secara rutin, dan memahami berbagai jenis ancaman siber.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

DAFTAR PUSTAKA

- Aayush, A., Aryan, Y., & Muniyal, B. (2022). Understanding SSL Protocol and Its Cryptographic Weaknesses. *2022 3rd International Conference on Intelligent Engineering and Management (ICIEM)*, 825–832. <https://doi.org/10.1109/ICIEM54221.2022.9853153>
- Alby, M. F., Ruslan, I. F., & Muharman, M. L. (2022). Information Security Test on Websites and Social Media Using Footprinting Method. *Proceedings of the 8th International Conference on Industrial and Business Engineering*. <https://doi.org/10.1145/3568834.3568868>
- Al-Shareeda, M., Anbar, M., Manickam, S., & Hasbullah, I. (2020). Review of Prevention Schemes for Man-In-The-Middle (MITM) Attack in Vehicular Ad hoc Networks. *International Journal of Engineering and Management Research*. <https://doi.org/10.31033/ijemr.10.3.23>
- Arman, M., & Rachmat, N. (2020). Implementasi sistem keamanan web server menggunakan pfsense. *Jusikom: Jurnal Sistem Komputer Musirawas*, 5(1), 13–23.
- Bhargavan, K., Bichhawat, A., Huy, Q., Hosseini, P., Küsters, R., Schmitz, G., & Würtele, T. (2021). An In-Depth Symbolic Security Analysis of the ACME Standard. *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*. <https://doi.org/10.1145/3460120.3484588>
- Chen, C., Tian, C., Duan, Z., & Zhao, L. (2018). RFC-Directed Differential Testing of Certificate Validation in SSL/TLS Implementations. *2018 IEEE/ACM 40th International Conference on Software Engineering (ICSE)*, 859–870. <https://doi.org/10.1145/3180155.3180226>
- Fattah, H. (2021). Hypertext Transfer Protocol (HTTP). *LTETM Cellular Narrowband Internet of Things (NB-IoT)*. <https://doi.org/10.1201/9781003120018-15>



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

- Hu, Q., Asghar, M., & Brownlee, N. (2021). A large-scale analysis of HTTPS deployments: Challenges, solutions, and recommendations. *J. Comput. Secur.*, 29, 25–50. <https://doi.org/10.3233/jcs-200070>
- Kampourakis, V., Kambourakis, G., Chatzoglou, E., & Zaroliagis, C. (2022). Revisiting man-in-the-middle attacks against HTTPS. *Netw. Secur.*, 2022. [https://doi.org/10.12968/s1353-4858\(22\)70028-1](https://doi.org/10.12968/s1353-4858(22)70028-1)
- Khan, N., Khan, A., Kar, H., Ahmad, Z., Tarmizi, S., & Julaihi, A. (2022). Employing Public Key Infrastructure to Encapsulate Messages During Transport Layer Security Handshake Procedure. *2022 Applied Informatics International Conference (AiIC)*, 126–130. <https://doi.org/10.1109/AiIC54368.2022.9914605>
- King, G., & Wang, H. (2021). *HTTPA: HTTPS Attestable Protocol*. 811–823. https://doi.org/10.1007/978-3-031-28073-3_54
- Koch, J., & Hao, W. (2021). Apache and HTTP/2 in the Cloud: A Comparative Study of Apache Architecture in AWS. *2021 IEEE 12th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, 673–680. <https://doi.org/10.1109/iemcon53756.2021.9623230>
- Kponyo, J., Agyemang, J., & Klogo, G. (2020). Detecting End-Point (EP) Man-In-The-Middle (MITM) Attack based on ARP Analysis: A Machine Learning Approach. *Int. J. Commun. Networks Inf. Secur.*, 12. <https://doi.org/10.22541/au.158938565.57807215>
- Liu, J., & Cheng, W. (2024). Design and Implementation of a Web Application Firewall System based on OpenResty. *2024 9th International Symposium on Computer and Information Processing Technology (ISCIPT)*, 6–9. <https://doi.org/10.1109/ISCIPT61983.2024.10673202>
- Mai, A., Schedler, O., Weippl, E., & Krombholz, K. (2022). Are HTTPS Configurations Still a Challenge?: Validating Theories of Administrators' Difficulties with TLS Configurations. 173–193. https://doi.org/10.1007/978-3-031-05563-8_12



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

- Mallik, A. (2019). MAN-IN-THE-MIDDLE-ATTACK: UNDERSTANDING IN SIMPLE WORDS. *Cyberspace: Jurnal Pendidikan Teknologi Informasi*. <https://doi.org/10.22373/cj.v2i2.3453>
- Momin, M. M., & Ali, O. (2023). Comprehensive Review of the Impact of Advanced Technology Adoption on Work and Continuous Improvement. *HighTech and Innovation Journal*, 4(3), 667–680. <https://doi.org/10.28991/HIJ-2023-04-03-014>
- Mundt, M., & Baier, H. (2023). Threat-Based Simulation of Data Exfiltration Toward Mitigating Multiple Ransomware Extortions. *Digital Threats: Research and Practice*, 4(4). <https://doi.org/10.1145/3568993>
- Nugroho, F. R., Afiana, F. N., & Kuncoro, A. P. (2024). NIST Cyber Security Framework Development for Website Information Collection. *Jurnal Teknologi Sistem Informasi Dan Aplikasi*, 7(3), 1335–1342. <https://doi.org/10.32493/jtsi.v7i3.41541>
- Nurlailah, E., & Nova Wardani, K. R. (2023). PERANCANGAN WEBSITE SEBAGAI MEDIA INFORMASI DAN PROMOSI OLEH-OLEH KHAS KOTA PAGARALAM. *JIPI (Jurnal Ilmiah Penelitian Dan Pembelajaran Informatika)*, 8(4), 1175–1185. <https://doi.org/10.29100/jipi.v8i4.4006>
- Nyabuto, G. (2023). Client-server Architecture, a Review. *International Journal of Advanced Science and Computer Applications*. <https://doi.org/10.47679/ijasca.v3i1.48>
- Poghosyan, I. (2024). ADVANTAGE OF INSTALLING MOODLE LEARNING MANAGEMENT SYSTEM UNDER IIS/MARIADB OVER APACHE/MYSQL. *Main Issues Of Pedagogy And Psychology*. <https://doi.org/10.24234/miopap.v11i2.49>
- Tripathy, S. S. (2024). A comprehensive survey of cybercrimes in India over the last decade. *International Journal of Science and Research Archive*. <https://doi.org/10.30574/ijusra.2024.13.1.1919>
- Westfall, J. (2021). *Basic Linux Administration via GUI (Webmin)*. 77–110. https://doi.org/10.1007/978-1-4842-6966-4_4



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Xia, K., Zhang, L., Yuan, S., & Lou, Y. (2023). SCSS: An Intelligent Security System to Guard City Public Safe. *IEEE Access*, 11, 76415–76426. <https://doi.org/10.1109/ACCESS.2023.3297643>

Zhang, X.-G., Yang, G., & Wasly, S. (2021). Man-in-the-middle attack against cyber-physical systems under random access protocol. *Inf. Sci.*, 576, 708–724. <https://doi.org/10.1016/j.ins.2021.07.083>

