# BAB I
# PENDAHULUAN

## 1.1. Latar Belakang

*File Transfer Protocol* atau biasa disingkat FTP adalah sebuah protokol yang mengatur bagaimana data, dalam hal ini berkas komputer, ditransfer dalam sebuah jaringan komunikasi digital. Beberapa tujuan dikembangkannya FTP antara lain adalah untuk mendukung sistem berbagi berkas dan kemudahan penggunaannya [1]. Mekanisme berbagi berkas atau *file transfer* menggunakan FTP masih digunakan dari awal lahirnya *internet* hingga saat ini, pada tahun 2015 diperkirakan ada belasan juta FTP server di alamat IP versi 4 di seluruh dunia [2].

Kemudahan penggunaan FTP tersebut pada umumnya menimbulkan pengorbanan pada aspek lain seperti keamanan data yang ditransmisikan dan yang disimpan di FTP server. Protokol ini tidak menyediakan mekanisme enkripsi atau perlindungan berkas data [3], sehingga beresiko berkas – berkas sensitif dapat diakses oleh pihak yang tidak bertanggung jawab, hal ini dibuktikan pada kasus - kasus kebocoran data pada FTP. Beberapa kasus kebocoran data dari FTP server beberapa dekade ke belakang menunjukkan bagaimana pentingnya aspek keamanan berkas yang dikirimkan menggunakan FTP, karena berpengaruh besar pada bidang pertahanan, bisnis dan pemerintahan—dimana sebagian besar informasi bersifat sensitif dan rahasia [4]. Di tahun 2018 miskonfigurasi pada FTP server sebuah perusahaan perangkat lunak menyebabkan ratusan ribu data pasien bocor [5], kemudian pada bulan juli tahun 2024 perusahaan negara milik India BSNL mengalami kebocoran data pada sistem FTP mereka sehingga membahayakan data pelanggan [6], di tahun yang sama pada bulan Oktober AFP melaporkan terjadinya kebocoran data yang disebabkan oleh peretasan kredensial FTP server sehingga memungkinkan akses ke konten - konten perusahaan media berita tersebut [7]. Dari kasus – kasus yang telah diuraikan sebelumnya, permasalahan utama yang terjadi adalah ketika *server* terkompromi, maka penyerang dapat dengan mudah mengakses konten dari berkas – berkas sensitif. Oleh karena itu, aspek keamanan atau kerahasiaan dan resiko kebocoran data pada berkas pada FTP *server* perlu diperhatikan secara serius. Salah satu metode yang bisa digunakan untuk

menyelesaikan masalah keamanan berkas yang tersimpan di FTP adalah dengan menggunakan Kriptografi.

Penelitian ini berfokus pada implementasi algoritma kriptografi kunci simeteri dengan jenis cipher alir untuk mengamankan data berupa berkas dokumen, yang selanjutnya hasil enkripsi berkas dikirim dan disimpan ke FTP/FTPs *server*. Implementasi dilakukan dengan merancang sebuah program menggunakan bahasa pemprograman Python. Selanjutnya disediakan dua jenis server sebagai simulasi melakukan enkripsi dan pengiriman data, dimana *server* pertama terkonfigurasi menggunakan FTP biasa dan *server* kedua menggunakan FTP dengan lapisan keamanan tambahan TLS atau FTPS.

Algoritma kriptografi simetris yang digunakan pada penelitian ini adalah *cipher* alir Salsa20 dengan *round* 20, algoritma ini dikembangkan oleh Daniel J. Bernstein pada tahun 2005, dengan panjang kunci 256-*bit* atau 32-*byte*, pengguna memasukan *secret key* dengan kombinasi huruf, angka dan karakter, selanjutnya fungsi hash SHA-256 digunakan untuk memudahkan agar *key* yang dimasukan pengguna berukuran 256-*bit* sesuai ketentuan panjang *bit* kunci Salsa20.

Pengujian dilakukan dengan menjalankan fungsionalitas program untuk mengetahui bahwa program berjalan dengan baik sesuai dengan desain yang telah ditetapkan, selanjutnya uji serangan *sniffing attack* dilakukan untuk melihat apakah berkas dokumen terenkripsi terjaga keamanannya, kemudian dilakukan analisis *known-plaintext attack* untuk menunjukkan bahwa *key* dan *nonce* tidak boleh digunakan kembali karena bisa mengorbankan kerahasiaan atau keamanan data, terakhir dilakukan perhitungan *avalanche effect* untuk memastikan bahwa pemilihan *cipher* sudah tepat dengan mempertimbangkan kekuatan *cipher* berdasarkan nilai presentase *avalanche effect*.

Salsa20 dipilih dengan mempertimbangkan beberapa alasan diantaranya Salsa20 tergolong algoritma cepat dibandingkan dengan algoritma lain yang biasa digunakan untuk enkripsi berkas, studi Park J dan Park E menunjukkan data dengan ukuran 0.5 GB – 10 GB algoritma Salsa20 melakukan proses enkripsi kurang lebih 3 kali dan 9 kali lebih cepat dibandingkan dengan algoritma AES dan Blowfish, serta 1.5 kali dan 8 kali lebih cepat dari kedua algoritma tersebut untuk proses

dekripsi [8], sedangkan penelitian J. Dawson, T. Frimpong dan J. Acquah Salsa20 menunjukkan nilai *throughput* (KB/detik) untuk proses enkripsi dan dekripsi lebih baik dibandingkan dengan algoritma varian Salsa20 yaitu ChaCha dan algoritma kunci publik RSA [9]. Pertimbangan terakhir adalah keamanan dari algoritma Salsa20 itu sendiri dimana sampai saat ini belum ada serangan yang mampu memecahkan kunci 256 *bit* pada *round* 20 atau *round* penuh, serangan terbaik sampai saat ini masih bersifat teoritis dimana pada tahun 2008 Aumasson et al. melaporkan dibutuhkan waktu komputasi sebesar $2^{153}$ untuk berhasil menyerang Salsa20 *round* 7[10] dan $2^{251}$ untuk *round* 8 [11], yang mana secara praktis jika menggunakan komputer modern saat ini masih tergolong tidak memungkinkan sehingga Salsa20 dikategorikan sebagai algoritma yang aman digunakan untuk beberapa waktu yang akan datang (*forseeable future*).

## 1.2. Rumusan Masalah

Berdasarkan latar belakang yang telah disampaikan sebelumnya, didapatkan rumusan masalah dari penelitian ini adalah sebagai berikut:

FTP masih digunakan sampai saat ini untuk berbagi berkas antara komputer, akan tetapi protokol ini tidak memiliki mekanisme pengamanaan atau enkripsi data bawaan baik ketika data tersebut ditransmisikan atauput disimpan di FTP *server,* oleh karena itu dibutuhkan sebuah metode untuk mengamankan atau menjaga kerahasiaan data berkas pada FTP tersebut dengan mengimplementasi algoritma kriptografi Salsa20 dalam bentuk perancangan program dilengkapi dengan pengujian fungsionalitas program, pengujian keamanan atau kerahasiaan berkas yang sudah terenkripsi, dan uji kelayakan atau kekuatan dari Salsa20 itu sendiri.

## 1.3. Batasan masalah

Adapun batasan masalah yang disajikan dalam penelitian ini adalah sebagai berikut:

1. Program perangkat lunak berbasis GUI ditulis dalam bahasa Python menggunakan paket GUI tkinter.
2. Program mendukung pengiriman dan enkripsi berkas untuk FTP dan FTPS.
3. Berkas yang akan dienkripsi adalah jenis berkas dokumen *txt, pdf*, word, *powerpoint*, dan *spreadsheet*.
4. Algoritma kriptografi yang diterapkan adalah Salsa20; kelompok algoritma kunci-simetri berjenis *cipher* alir.
5. Proses enkripsi dan dekripsi berkas dilakukan menggunakan paket *Python PyCryptodome*.
6. Proses koneksi FTP pada program dilakukan menggunakan *paket Python ftplib*.
7. Dua FTP server dibuat untuk melakukan simulasi enkripsi dan pengiriman berkas.
8. FTP *server* dipasang pada sistem operasi linux Ubuntu yang dijalankan di layanan *Virtual Machine* Microsoft Azure Cloud.

## 1.4. Tujuan

Berdasarkan rumusan masalah yang ada, maka tujuan dari penelitian yang dilakukan adalah sebagai berikut:

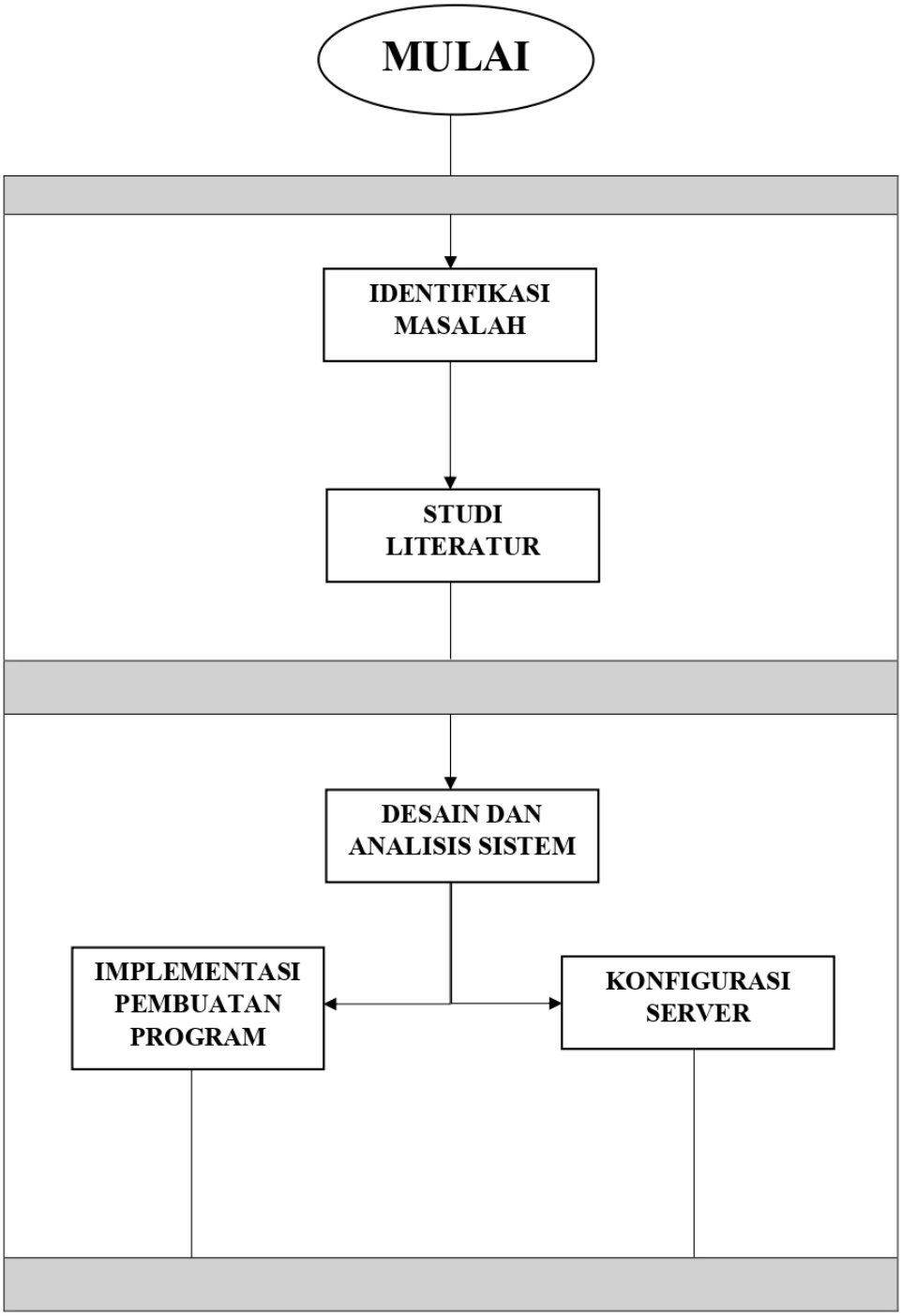
1. Merancang sebuah program berbasis GUI untuk mengamankan atau menjaga kerahasiaan (*confidentiality*) berkas dokumen pada FTP server menggunakan algoritma kriptografi Salsa20.
2. Menguji fungsionalitas program, keamanan atau kerahasiaan berkas dokumen yang sudah terenkripsi, dan kekuatan algoritma kriptografi.

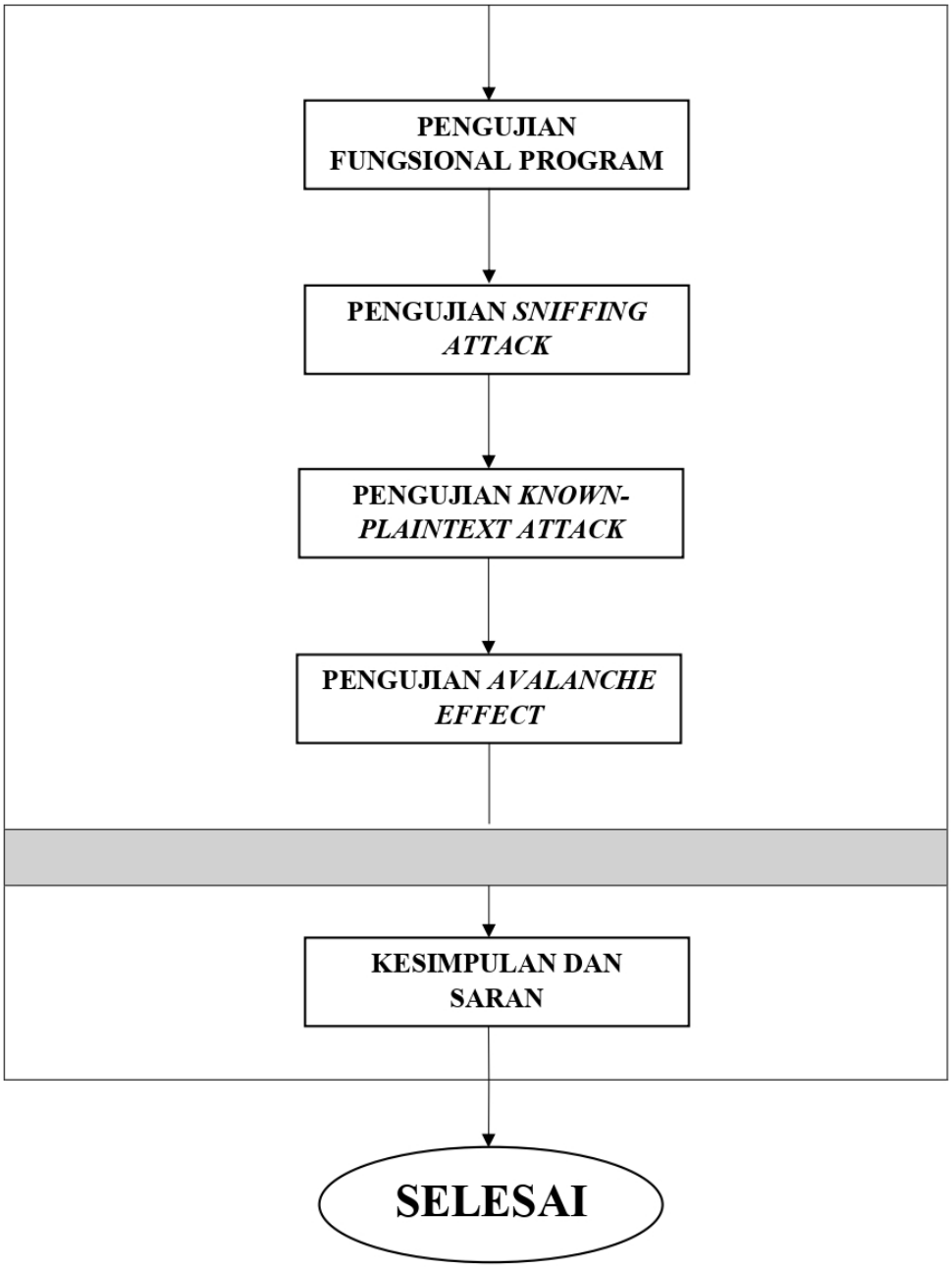
## 1.5. Manfaat

Mengetahui bagaimana algoritma kriptografi Salsa20 diterapkan untuk mengamankan data digital—dalam hal ini berkas dokumen—yang disimpan ke FTP server, sehingga menjaga kerahasiaan data *at rest*. Selanjutnya penelitian ini bermanfaat sebagai pertimbangan dan rujukan metode enkripsi kriptografi menggunakan algoritma Salsa20 untuk mengamankan data berkas dokumen yang tersimpan di suatu media penyimpanan.

## 1.6. Rencana Kegiatan

Agar penelitian lebih terstruktur dan terarah, maka dibuat diagram alir rencana kegiatan. Tahap – tahap penelitian dipisah berdasarkan kegiatan seperti identifikasi masalah, studi literatur, perancangan, pengujian dan analisis dan penyimpulan hasil. Gambar 1.1 menunjukkan diagram alur renjana kegiatan penelitian.






*Gambar 1.1 Alur Penelitian*

### 1.6.1. Identifikasi Masalah

Secara umum tahapan ini menentukan tema penelitian sebagai acuan dasar tahapan berikutnya. Setelah tema ditetapkan, selanjutnya mengidentifikasi dan merumuskan masalah. Rumusan masalah berisi penjelasan masalah terkait dengan topik penelitian yang akan dijawab peneliti

menggunakan solusi dan metode yang ditawarkan. Berdasarkan rumusan masalah yang ada kemudian diperoleh pertanyaan dan tujuan untuk mengetahui arah dan fokus penelitian.

### 1.6.2. Studi Literatur

Studi literatur yang berisi ulasan atau kajian teoritis metode atau solusi untuk menyelesaikan masalah penelitian, literatur rujukan didapatkan dari berbagai sumber seperti buku, jurnal, artikel, laporan, dan *website*, tahap ini juga meliputi penelitian - penelitian terdahulu yang berisi metode penyelesaian masalah sejenis sebagai sumber untuk mengidentifiksi potensi kesenjangan pengetahuan serta memperjelas posisi penelitian untuk mengisi gap tersebut.

### 1.6.3. Perancangan

Kegiatan perancangan terbagi menjadi tiga tahap diantaranya desain dan analis sistem, implementasi pembuatan program, dan konfigurasi server.

Pada tahap desain dan analisis sistem dibuat sebuah rancangan skenario interasksi pengguna dengan perangkat lunak dan sistem untuk mencapai suatu tujuan, dijelaskan juga fitur-fitur dari perangkat lunak yang akan dikembangkan. Perancangan dilakukan dengan menggunakan bantuan *Unified Modelling Language* (UML), diagram UML yang akan digunakan adalah *Use Case* dan *Activity*. Setelah proses perancangan *Use Case* dilanjutkan proses merancang *Activity Diagram*. *Activity Diagram* menampilkan alir langkah demi langkah aktivitas pengguna pada sistem di setiap fitur program, pada proses ini juga dijelaskan setiap langkah tersebut.

Pada tahap implementasi pembuatan program dilakukan pengembangan program sesuai dengan fitur – fitur yang sebelumnya telah dijelaskan di tahap desain dan analisis sistem, proses utama pada tahap ini adalah penulisan kode sumber dengan koleksi atau pustaka pendukung sesuai kebutuhan program yang akan dibangun.

Tahapan terakhir pada kegiatan perancangan adalah konfigurasi server sesuai dengan kebutuhan yang akan digunakan untuk mensimulasikan pengiriman berkas ke sebuah *server*.

### 1.6.4. Pengujian dan Analisis

Kegiatan pengujian dan analisis meliputi empat tahapan diantaranya pengujian fungsionalitas program, pengujian *sniffing attack*, pengujian *known-plaintext attack* dan pengujian *avalanche effect*.

### 1.6.5. Penyimpulan Hasil

Kegiatan terakhir berisi satu tahapan yaitu kesimpulan dan saran. Pada tahap ini dilakukan penarikan kesimpulan dari hasil perancangan dan pengujian, serta saran peneliti untuk memperbaiki penelitian.