**A**

**Project**

**On**

<STEGANOGRAPHY TOOL >

**For the partial fulfilment of the Course**

***Linux Shell Programming (CSET-213/CBCA221)***

**Academic Year: 2024-24 (Odd Semester)**

**Submitted by**

<LUCKYCHOWDARY >

**Under The Supervision of**

Dr Vimal Kumar/Dr. Debjani Ghosh/Dr Nitin Shukla

School of Computer Science Engineering & Technology

Bennett University (The Times Group), Greater Noida, UP-201310, India

****

**SCHOOL OF COMPUTER SCIENCE ENGINEERING & TECHNOLOGY**

**BENNETT UNIVERSITY (THE TIMES GROUP),**

**GREATER NOIDA, UP-201310, INDIA**

**Contents**

|  |  |
| --- | --- |
| ***Topic*** | ***Page No*** |
| Introduction | **3** |
| Background | **3** |
| Project Scope | **3** |
| Addressed the Problem: Targeted Applications | **4** |
| Project Objectives | **4** |
| Software Requirement Specification (SRS) & Playbook Design: Functional & non-functional Requirements | **4** |
| Implementation | **5** |
| Resources Used | **5** |
| Phases and Milestones | **6** |
| Conclusion | **6** |

**Got it! I'll draft a project report for your steganography tool, modeled after the structure and tone of the shared demo. Here's a detailed version of the report:**

**Project Report: File Steganography Tool**

**1. Introduction**

**In the digital age, the importance of secure communication and data protection has increased significantly. Steganography, the art of concealing information within other files, has emerged as a powerful tool to ensure confidentiality and prevent unauthorized access. This project presents a File Steganography Tool that allows users to hide sensitive files within .png images, ensuring that the concealed data remains undetectable to casual inspection.**

**The tool is designed to provide a secure and user-friendly solution for individuals and organizations seeking enhanced data protection. By embedding files within image files and securing them with encryption, the tool ensures both confidentiality and accessibility, even in sensitive scenarios.**

**2. Background**

**Traditional file-sharing methods often expose sensitive data to interception and unauthorized access. Encryption offers protection, but encrypted files can attract unwanted attention due to their distinct structure. Steganography addresses this issue by concealing the presence of sensitive files within ordinary images.**

**This project aims to bridge the gap between encryption and secure data hiding by developing a steganography tool capable of embedding and extracting files using password-protected encryption. The tool supports .png format as the carrier and can conceal text or files (e.g., .txt, .png) within it.**

**The goal is to offer a robust yet simple tool for secure file sharing, especially in environments where privacy and security are paramount.**

**3. Project Scope**

**The scope of the File Steganography Tool includes:**

* **Embedding Files: The ability to hide sensitive data (text or files) within .png images.**
* **Extraction of Hidden Data: A mechanism to securely retrieve embedded files using a password.**
* **Encryption: Securing hidden data with AES-256 encryption to ensure unauthorized access is prevented.**
* **Ease of Use: A user-friendly interface for embedding and extracting data.**

**The tool is designed for a wide range of applications, including:**

* **Secure Communication: Enabling individuals to share sensitive information without raising suspicion.**
* **Data Protection: Concealing critical files in unassuming image formats to protect against unauthorized access.**
* **Educational Use: Demonstrating the principles of steganography and secure data embedding.**

**4. Addressed Problems and Targeted Applications**

**The File Steganography Tool addresses the following challenges:**

* **Secure File Sharing: Preventing unauthorized access during data transmission.**
* **Enhanced Privacy: Concealing the existence of sensitive files within ordinary images.**
* **User-Friendly Security: Providing an intuitive interface for non-technical users to securely embed and extract files.**

**Targeted Applications**

* **Secure file transmission over untrusted networks.**
* **Concealing sensitive information in public communication.**
* **Educational demonstrations of steganographic techniques.**

**5. Project Objectives**

**The primary objectives of the File Steganography Tool include:**

* **Integration of Encryption: Use AES-256 encryption for secure data storage.**
* **File Embedding and Extraction: Enable users to embed and retrieve files within .png images seamlessly.**
* **Cross-Platform Compatibility: Ensure the tool works efficiently on Linux-based systems.**
* **Ease of Use: Provide a simple interface for both technical and non-technical users.**
* **Reliability: Ensure the integrity of embedded files during embedding, extraction, and transmission.**

**6. Software Requirements and Playbook Design**

**Functional Requirements**

* **File Embedding: Allow users to embed files within .png images.**
* **File Extraction: Retrieve embedded files with password authentication.**
* **Encryption: Use AES-256 encryption to secure hidden files.**
* **Format Support: Support .txt and .png as input/output formats.**

**Non-Functional Requirements**

* **Security: Encrypt data before embedding to ensure confidentiality.**
* **Performance: Efficiently handle file embedding and extraction without high resource consumption.**
* **Usability: Provide a clear and intuitive command-line interface.**
* **Compatibility: Support common Linux distributions.**

**7. Implementation**

**The File Steganography Tool was implemented using Bash scripting for its simplicity and compatibility across Linux systems. Key features include:**

* **File Embedding: The tool uses a combination of openssl for encryption and cat for appending encrypted data to the carrier file.**
* **File Extraction: Extracts and decrypts the hidden file based on unique markers embedded during the hiding process.**
* **Encryption: AES-256 encryption ensures hidden files are secure, even if extracted maliciously.**

**Technologies Used**

* **Bash: For scripting the embedding and extraction processes.**
* **OpenSSL: For encrypting and decrypting hidden data.**
* **AWK: For identifying and extracting hidden data based on unique markers.**
* **Linux Utilities: Core tools like cat, echo, and read for file handling.**

**8. Resources Used**

**Software**

* **Bash Script: For the implementation of steganographic operations.**
* **OpenSSL: For encryption and decryption.**
* **Linux OS: A compatible environment for running the tool.**

**Hardware**

* **Standard desktop or server hardware running a Linux-based operating system.**

**9. Phases and Milestones**

**The project was developed in the following phases:**

1. **Planning and Design: Defining the requirements and identifying suitable steganographic techniques.**
2. **Core Development: Implementing file embedding and extraction functionality with encryption.**
3. **Testing: Ensuring compatibility, reliability, and security under various conditions.**
4. **Optimization: Enhancing the tool’s performance and usability.**

**10. Conclusion**

**The File Steganography Tool provides a robust solution for secure data hiding and retrieval. By combining steganography with encryption, it ensures that sensitive information remains protected while blending seamlessly into everyday file formats. The tool demonstrates the potential of steganography for secure communication and data protection in an increasingly connected world.**