National College of Ireland

Configuration Manual

MSc Research Project
MSc In Cyber Security

Nishanth Marisamy
Student ID: X23331968

School of Computing
National College of Ireland

Supervisor: Eugene Mclaughlin

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Nishanth Marisamy
Student ID:	X23331968
Programme:	MSc In Cyber Security **Year:** 2025-2026
Module:	MSc in Cyber Security
Lecturer:	Eugene Mclaughlin
Submission Due Date:	11/12/2025
Project Title:	A Hybrid ChaCha20-Based Lightweight Cryptographic Framework for Latency and Diffusion Optimization in IoT Systems
Word Count:	669 **Page Count: 4**

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Nishanth Marisamy
Date:	11/12/2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

Nishanth Marisamy
Student ID: X23331968

1. Introduction

IoT environments rely on devices with limited processing power, memory, and energy capacity, making traditional cryptographic systems unsuitable due to high latency and restricted diffusion strength. This project introduces a hybrid encryption framework that combines lightweight ciphers (SPECK, SIMON) for device-level processing with robust ciphers (AES or ChaCha20) executed at the cloud layer. By integrating parallelized encryption, optimized round processing, and cloud offloading, the system achieves improved security, lower latency, and stronger diffusion. The architecture is designed to suit real-time IoT operations while ensuring minimal computational overhead and enabling secure communication between IoT nodes and cloud services.

2. Hardware Requirements

CPU: Dual-core 2.0 GHz
RAM: 8 GB
Storage: 20 GB
Internet: Stable broadband connection

3. Software Requirements

Operating System: Windows 10/11
Python Version: 3.8 or above
Python Libraries: pycryptodome, cryptography, numpy, pandas, requests, flask
IDE: VS Code / PyCharm

4. Python 3.10.12

It is also a very generalized and easily manageable programming language which any new programmers as well as the professionals can use. It has a less complex grammar through which the programmer can express a single idea or plan of action in fewer lines of code than it is possible to do in other languages, this makes it more precise and effective. I found that these are procedural, object oriented and functional and it has support a number of libraries and frameworks for Web, data analysis, AI and many more. Because of this flexibility and

reliability, the scripts have been sought frequently in the spheres of activity including software development and scientific analysis.

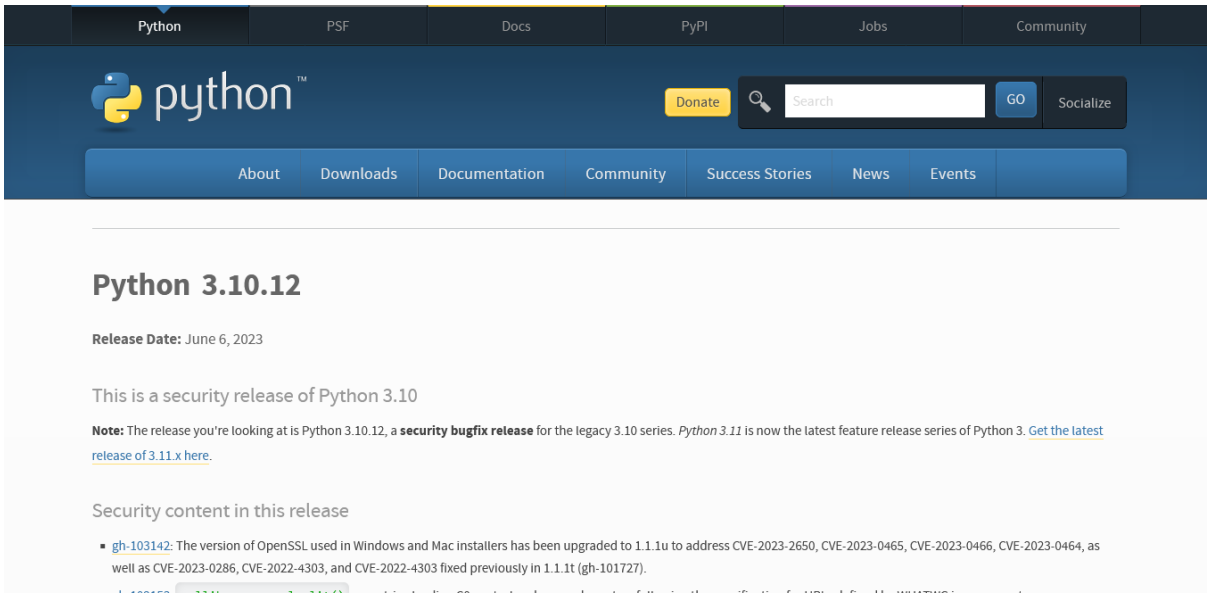

Fig : 1 Python 3.10.12

5. Visual Studio Code (VS Code)

Visual Studio Code (VS Code) – is a free code editor that is relatively simple and efficient, which is, in fact, developed by Microsoft. Very often it is used due to the high number of features available, especially the option to work in various programming languages through extensions. It is also stable for code suggestion, various debuggers, the appropriate terminal, and the integration of GitHub, which make the development very convenient in VS Code. General trading platform offers call for a vast number of extensions and themes that improve rendering for developers and organization of work on them. Due to these and other daily updates and users' interactions, VS Code is one of the most useful IDEs that developers can employ while working within the present conditions.

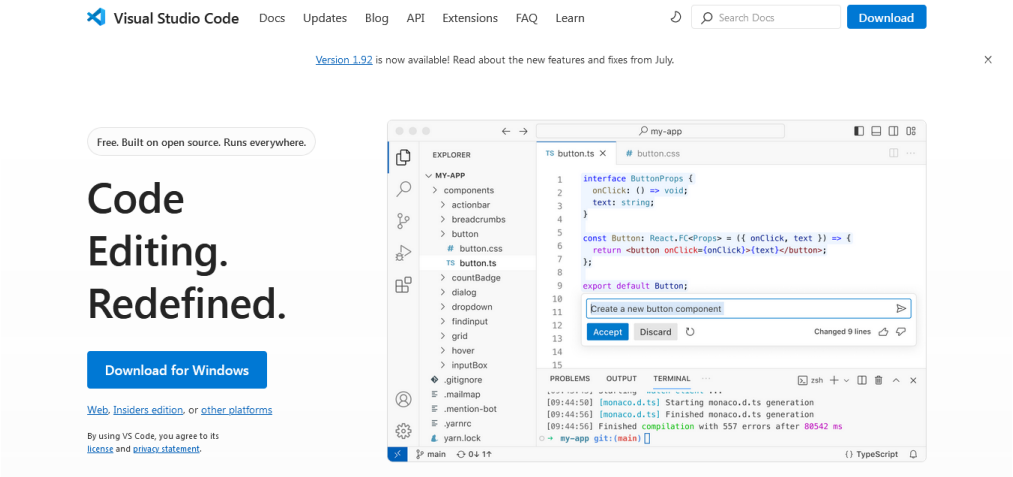

Fig 2 : VS Code

6. Technologies Used

- Cryptographic Algorithms
- ChaCha20 – Strong and fast stream cipher for cloud-level security
- AES – Industry-standard block cipher with strong diffusion
- SPECK / SIMON – Lightweight ciphers optimized for IoT devices
- Programming & Frameworks
- Python – Core language for implementation
- Flask – For building cloud encryption endpoints
- Parallel Processing
- Python multiprocessing / multithreading
- Block-level parallel encryption for latency reduction

6. Tools & Libraries

These are the tools and libraries that have been used in this project.

- PyCryptodome : Provides AES, ChaCha20, hashing, random generation, and secure crypto primitives. Used for robust cloud-layer encryption.

- Cryptography : Offers ECC, ChaCha20, AES, and secure key-handling features. Supports reliable and modern implementations.

- NumPy : Used for fast numerical operations, block handling, array transformations, and performance calculations.

- Pandas : Used to record, analyze, and compare performance metrics such as latency, throughput, and diffusion quality.

- Requests : Enables IoT-side Python client to communicate with cloud endpoints (sending encrypted data to EC2/Lambda).

- Flask : Used to build cloud service APIs for receiving data, re-encrypting with AES/ ChaCha20, and returning results.

- Multiprocessing / Threading : Allows parallel encryption to reduce latency and improve throughput.

7.Implementation Flow

- Use python for all the modules :
    - Implement SPECK, SIMON (or use existing lightweight libraries)
    - Use pycryptodome or cryptography library for AES and ChaCha20
- Parallel processing Integration
    - Implement multithreading or multiprocessing for encryption tasks in the cloud.
    - Split data into blocks for parallel encryption and reassembly.
    - Measure improvements in latency and throughput.
- Performance Evaluation
    - Test on various metrics:
        - Latency
        - CPU
        - Throughput
        - Avalanche effect (diffusion quality)
        - Encryption/decryption time
    - Compare:
        - Hybrid LWE system vs. standalone AES system vs ChaCha20 system

7. Output Screenshots

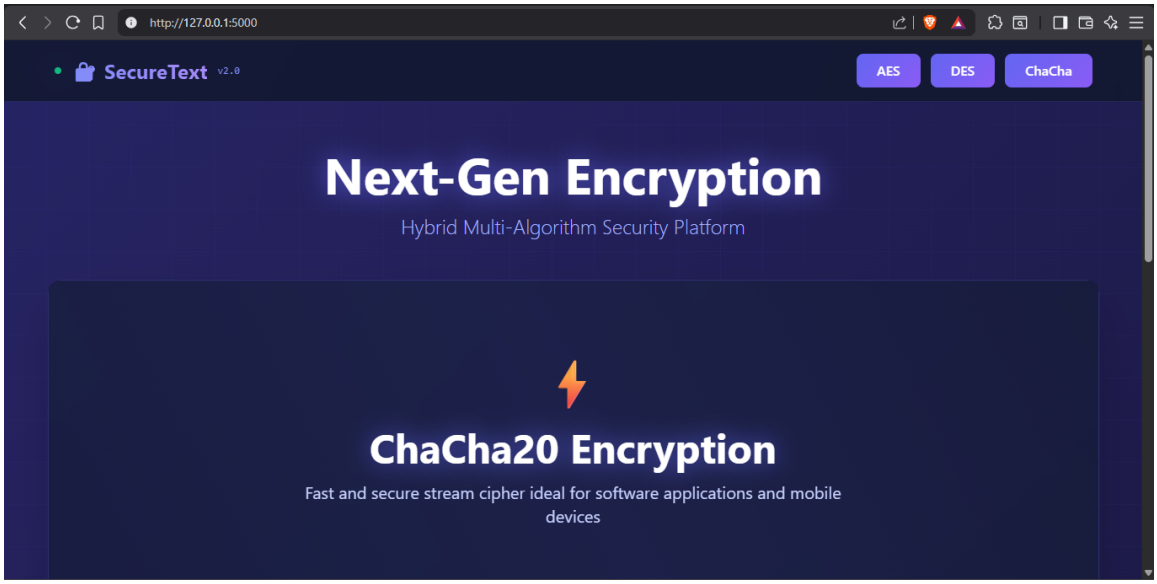

Fig 3 : OUTPUT-1

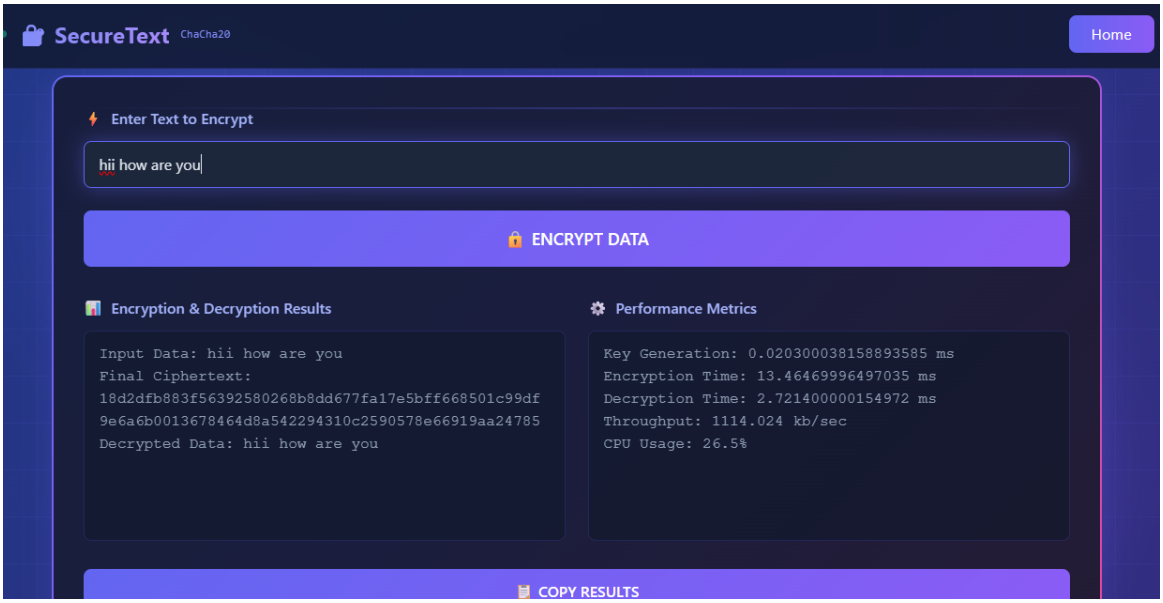

Fig 4 : OUTPUT-2