

Chunk-Link File Integrity Tool - Documentation

1. Introduction

The Chunk-Link File Integrity Tool is a Python-based application designed to split files into smaller chunks, establish a chain of integrity using cryptographic checksums, and validate the integrity of files. This tool is essential for ensuring data hasn't been tampered with or corrupted during storage or transfer.

2. Core Functionality

2.1 Chunking and Linking Mechanism

- **Chunking:** Files are divided into smaller, manageable pieces called chunks.
- **Integrity Chain:** Each chunk is cryptographically linked to the next by including the checksum of the subsequent chunk. Any modification to a chunk will break this chain.
- **Checksums:** SHA-256 hashing algorithm is used to generate unique fingerprints (checksums) for each chunk, ensuring even the smallest change is detectable.

2.2 Operations

- **Split:** Divides a selected file into chunks of a specified size and creates the integrity chain.
- **Reconstruct:** Reassembles the original file from its chunks using the integrity chain.
- **Validate:** Checks the integrity of the file by traversing the chain and comparing checksums.
- **Simulate Corruption:** Intentionally alters a chunk to demonstrate the tool's ability to detect corruption.

3. How to Use

1. **Select File:** Choose a file to process.
2. **Chunk Size:** Define the size of each chunk (default is 1KB).
3. **Split:** Click "Split" to divide the file and create the integrity chain.
4. **Reconstruct:** Choose an output location and click "Reconstruct" to rebuild the original file.
5. **Validate:** Click "Validate" to confirm the file's integrity.
6. **Simulate Corruption:** Use this function to test the tool's detection capabilities.

4. Display and Information

The tool provides a display area to show:

- **Chunk Index:** Sequential number of each chunk.
- **Data Snippet:** A preview of the chunk's data in hexadecimal format.
- **Next Checksum:** The SHA-256 checksum of the subsequent chunk, linking it to the chain.

5. Example

1. **File:** "document.txt" is split into 3 chunks.
2. **Chunk 1:** Contains the first part of "document.txt" and the checksum of Chunk 2.
3. **Chunk 2:** Contains the middle part of "document.txt" and the checksum of Chunk 3.
4. **Chunk 3:** Contains the last part of "document.txt".
5. **Reconstruction:** The file is rebuilt by concatenating chunks in order, verified by checksums.
6. **Validation:** Ensures that each chunk's data matches its checksum and the chain is unbroken.

6. Notes

- This tool is essential for verifying the integrity of stored or transferred files.
- Any change to the file, however slight, will be detected.
- The tool is user-friendly with a straightforward interface for file processing and validation.