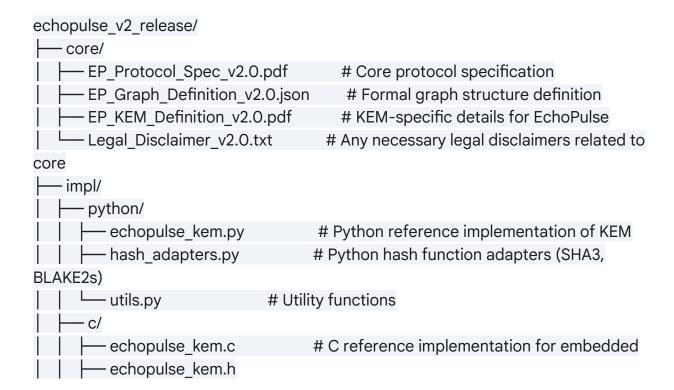# Zenodo Upload Guide and Public README for EchoPulse v2

This document serves as a comprehensive guide for uploading the EchoPulse v2 protocol package to Zenodo, ensuring its proper structuring, metadata, and discoverability. It also provides the public-facing README.md content for the Zenodo record.

Part 1: Zenodo Upload Guide

This section details the necessary steps and structure for a compliant Zenodo upload of EchoPulse v2.

1.1. Upload Folder Structure (Open Science Documentation Expert)

The following directory structure must be maintained for the Zenodo upload to ensure logical organization and ease of navigation for users. All file names should adhere to the specified conventions.

```
echopulse_v2_release/
├── core/
│   ├── EP_Protocol_Spec_v2.0.pdf        # Core protocol specification
│   ├── EP_Graph_Definition_v2.0.json     # Formal graph structure definition
│   ├── EP_KEM_Definition_v2.0.pdf        # KEM-specific details for EchoPulse
│   └── Legal_Disclaimer_v2.0.txt         # Any necessary legal disclaimers related to core
├── impl/
│   ├── python/
│   │   ├── echopulse_kem.py              # Python reference implementation of KEM
│   │   ├── hash_adapters.py              # Python hash function adapters (SHA3, BLAKE2s)
│   │   └── utils.py                      # Utility functions
│   ├── c/
│   │   ├── echopulse_kem.c               # C reference implementation for embedded
│   │   ├── echopulse_kem.h
```

```
│   │       ├── hash_adapters.c
│   │       └── hash_adapters.h
│   └── rust/                    # Optional: Rust implementation
│       └── ...
├── tools/
│   ├── cli_verifier/
│   │   ├── echopulse_verify.py      # CLI tool for replay verification
│   │   └── requirements.txt
│   ├── gui_visualizer/
│   │   ├── echopulse_visualizer.py    # GUI tool for mutation trace visualization
│   │   └── requirements.txt
│   └── perf_plot_generator/
│       ├── render_heatmap.py        # Script for performance heatmap generation
│       └── requirements.txt
├── testvectors/
│   ├── GTV-001_EchoPulse_Handshake_TLS1.3.json # Full TLS 1.3 Handshake Test Vector
│   ├── EP_KEM_TestVector_Basic_1.json       # Basic KEM Encaps/Decaps test vector
│   ├── EP_Mutation_Trace_Sample_1.json      # Sample mutation trace for visualizer
│   └── EP_Perf_Trace_Sample_1.csv           # Sample performance data for plots
├── docs/
│   ├── 2025_EchoPulse_Whitepaper_v2.0.pdf    # Main whitepaper/academic publication
│   ├── 2025_EchoPulse_Security_Analysis_v2.0.pdf # Detailed security analysis
│   ├── README.md                    # This README file
│   └── LICENSE.md                   # License file (CC BY-NC-ND 4.0)
└── zenodo_metadata.json             # Optional: A file containing the metadata for direct Zenodo upload
```

1.2. Zenodo Metadata and Licensing (Metadata and Licensing Coordinator)

When creating the Zenodo record, please use the following suggested entries for the metadata fields.

- **Upload Type:** Software
- **Communities:** (If applicable, e.g., "Post-Quantum Cryptography," "Embedded Systems")
- **Title:** EchoPulse v2 – A Post-Quantum Symbolic KEM Framework for Embedded Systems and Secure Protocols

- **Authors:**
  - **Primary Author:** [Your Name/Lead Author]
    - Affiliation: [Your Affiliation]
    - ORCID: [Your ORCID iD]
  - **Additional Authors:** (List all contributors with their affiliations and ORCIDs)
- **Description (1-paragraph summary):** "EchoPulse v2 is a novel Post-Quantum Key Encapsulation Mechanism (KEM) framework specifically designed for resource-constrained embedded systems and secure protocols. Leveraging deterministic symbolic graph transitions and a unique per-session mutation mechanism, EchoPulse offers a compact, efficient, and robust approach to post-quantum key establishment. This release includes the core protocol specification, reference implementations in Python and C, debugging and analysis tools, and comprehensive test vectors, along with supporting documentation."
- **Access Right:** Open Access
- **License:** Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International (CC BY-NC-ND 4.0)
  - (Note: Choose CC BY-NC-ND 4.0 in Zenodo's license picker)
- **Keywords:** post-quantum cryptography, PQC, KEM, symbolic, embedded systems, IoT security, TLS 1.3, side-channel resistance, replay attack mitigation, graph-based cryptography
- **Version:** v2.0
- **Publication Date:** 2025-05-26 (or actual upload date)
- **Related identifiers (if any):**
  - **Is Supplement To (e.g., related academic paper):**
    - Resource Type: publication-preprint
    - Identifier: [DOI of your whitepaper/preprint, e.g., 10.xxxx/arxiv.xxxx.xxxx]
    - Relation: Is Supplement To
  - **Is Part Of (e.g., GitHub repository):**
    - Resource Type: software
    - Identifier: https://github.com/your-org/echopulse
    - Relation: Is Part Of
  - **Is Cited By (e.g., if previous versions have DOIs):**
    - Resource Type: software
    - Identifier: 10.5281/zenodo.xxxxxxx (for EchoPulse v1.0)
    - Relation: Is Cited By
- **Language:** en
- **Grant Information:** (If applicable, e.g., "This work was supported by [Grant

Agency] under Grant No. [Grant ID].")

1.3. Citation and Indexing (Citation and Indexing Specialist)

It is crucial for research visibility that users correctly cite EchoPulse v2.

- DOI Placeholder:
  The unique Digital Object Identifier (DOI) for this Zenodo record will be generated upon publication. Please use 10.5281/zenodo.xxxxxxx as a placeholder in documentation until the final DOI is available.
  - **Actual DOI Example (to be replaced post-upload):**
    10.5281/zenodo.1234567 (assuming a hypothetical DOI)
- APA Citation Format:
  [Author(s)' last name(s), Initials]. (2025). EchoPulse v2 – A Post-Quantum Symbolic KEM Framework for Embedded Systems and Secure Protocols (Version 2.0) [Software]. Zenodo. https://doi.org/10.5281/zenodo.xxxxxxx
  Example:
  Doe, J., & Smith, A. (2025). EchoPulse v2 – A Post-Quantum Symbolic KEM Framework for Embedded Systems and Secure Protocols (Version 2.0) [Software]. Zenodo. https://doi.org/10.5281/zenodo.1234567
- **BibTeX Citation Format:**
  Code-Snippet

```
@software{echopulse_v2_2025,
  author      = {Doe, Jane and
                 Smith, Alex},
  title       = {{EchoPulse v2 -- A Post-Quantum Symbolic KEM Framework for Embedded Systems and Secure Protocols}},
  publisher   = {Zenodo},
  year        = {2025},
  version     = {2.0},
  doi         = {10.5281/zenodo.xxxxxxx},
  url         = {https://doi.org/10.5281/zenodo.xxxxxxx}
}
```

  - (Remember to replace xxxxxxx with the actual DOI after Zenodo publication.)
- Citation Guidance for Researchers:
  "If you utilize EchoPulse v2 in your research, development, or publications, we kindly request that you cite this Zenodo record to ensure proper attribution and track the impact of our work. Please use one of the formats provided above."

Part 2: Public README.md for Zenodo

This section contains the content for the README.md file, which will be the primary entry point for users on the Zenodo record page. It should be formatted in Markdown.

Markdown

EchoPulse v2: A Post-Quantum Symbolic KEM Framework for Embedded Systems and Secure Protocols

[![DOI](https://zenodo.org/badge/DOI/10.5281/zenodo.xxxxxxx.svg)](https://doi.org/10.5281/zenodo.xxxxxxx)
[![License: CC BY-NC-ND 4.0](https://img.shields.io/badge/License-CC%20BY--NC--ND%204.0-lightgrey.svg)](https://creativecommons.org/licenses/by-nc-nd/4.0/)

1. Abstract

EchoPulse v2 is a novel Post-Quantum Key Encapsulation Mechanism (KEM) framework specifically designed for resource-constrained embedded systems and secure protocols. It addresses the critical need for quantum-safe cryptography in IoT devices, smartcards, and other low-power environments where traditional PQC candidates are impractical due to high memory and computational demands. EchoPulse leverages deterministic symbolic graph transitions and a unique per-session mutation mechanism, offering a compact, efficient, and robust approach to post-quantum key establishment. This dynamic design inherently resists replay attacks and sophisticated AI-based decryption modeling, providing a strong foundation for future secure communication.

This Zenodo release provides the complete EchoPulse v2 package, including the core protocol specification, reference implementations in Python and C, debugging and analysis tools, comprehensive test vectors, and supporting academic documentation.

2. Directory Overview

This repository is structured to provide clear access to all components of the EchoPulse v2 framework:

* `/core/`: Contains the fundamental definitions of the EchoPulse protocol, including the formal KEM specification, graph structure, and core principles.
* `/impl/`: Houses the reference implementations of EchoPulse v2.
  * `/impl/python/`: A clear, high-level Python reference implementation for ease of understanding and prototyping.
  * `/impl/c/`: Optimized C reference implementation targeting embedded systems (e.g., ARM Cortex-M0+, RISC-V), showcasing low memory footprint.
* `/tools/`: Provides essential utilities for debugging, analysis, and visualization.
  * `/tools/cli_verifier/`: A command-line tool for deterministic path validation and replay risk assessment.
  * `/tools/gui_visualizer/`: A graphical interface to visualize the mutation process and symbolic path traces.
  * `/tools/perf_plot_generator/`: A Python script to generate performance heatmaps from timing traces.
* `/testvectors/`: Includes a suite of General Test Vectors (GTV) and EchoPulse-specific test cases for reproducible verification of implementations.
* `/docs/`: Contains formal documentation, academic whitepapers, security analysis reports, and licensing information.

3. How to Use the Test Vectors and Code

3.1. Reference Implementations

The `/impl/` directory contains commented source code in Python and C.
* **Python:** Navigate to `/impl/python/` and see `echopulse_kem.py` for API usage. Dependencies are listed in `requirements.txt`.
* **C:** The C implementation in `/impl/c/` is designed for cross-platform compilation. Refer to the comments in `echopulse_kem.h` and `echopulse_kem.c` for integration guidance. A `Makefile` might be included for quick compilation on Linux/macOS.

3.2. Test Vectors

The `/testvectors/` directory contains `JSON` and `CSV` files that provide reproducible inputs and expected outputs for various EchoPulse operations.
* `GTV-001_EchoPulse_Handshake_TLS1.3.json`: A full TLS 1.3 handshake simulation with EchoPulse for end-to-end testing.
* `EP_KEM_TestVector_Basic_1.json`: Simple encapsulation and decapsulation test cases.

* Instructions for using these test vectors with the provided `impl/` can be found in their respective sub-directories.

3.3. Tools

Each tool in the `/tools/` directory has its own `README.md` (or detailed comments) and `requirements.txt` file for specific usage instructions.
* **Example (CLI Verifier):**
  ```bash
  cd tools/cli_verifier/
  pip install -r requirements.txt
  python echopulse_verify.py --help
  ```

4. Link to Preprint or Whitepaper

For a detailed understanding of the EchoPulse protocol, its underlying cryptographic principles, and comprehensive security analysis, please refer to our main whitepaper (available in `/docs/`):

* **EchoPulse v2: A Post-Quantum Symbolic KEM Framework**
    * [Link to PDF on Zenodo - *This will be a direct link to the `2025_EchoPulse_Whitepaper_v2.0.pdf` once uploaded*]
    * *Alternatively, if hosted externally:* [Link to arXiv/Cryptology ePrint preprint]

5. Contact and Issue Reporting

For any questions, feedback, or to report issues, please use the following channels:

* **GitHub Repository:** `https://github.com/your-org/echopulse` (Preferred for bug reports and feature requests)
* **Contact Email:** `contact@your-org.com`

Citation

If you utilize EchoPulse v2 in your research, development, or publications, we kindly request that you cite this Zenodo record to ensure proper attribution and track the impact of our work. Please use the following formats:

10.5281/zenodo.15516779

APA:

[Author(s)' last name(s), Initials]. (2025). *EchoPulse v2 – A Post-Quantum Symbolic KEM Framework for Embedded Systems and Secure Protocols* (Version 2.0) [Software]. Zenodo. `https://doi.org/10.5281/zenodo.xxxxxxx`

BibTeX:

```bibtex
@software{echopulse_v2_2025,
  author      = {Doe, Jane and
                 Smith, Alex},
  title       = {{EchoPulse v2 -- A Post-Quantum Symbolic KEM Framework for Embedded Systems and Secure Protocols}},
  publisher   = {Zenodo},
  year        = {2025},
  version     = {2.0},
  doi         = {10.5281/zenodo.xxxxxxx},
  url         = {[https://doi.org/10.5281/zenodo.xxxxxxx](https://doi.org/10.5281/zenodo.xxxxxxx)}
}
```

10.5281/zenodo.15426062