EchoPulse Symbol Mapping Reference

1. Definition of Σ

The EchoPulse protocol utilizes a finite set of 256 distinct, abstract symbols, denoted as $\Sigma=\{s0,s1,...,s255\}$. These symbols are the fundamental units of operation within the state transition graph G(V,E). They are explicitly defined as non-algebraic entities, possessing no inherent mathematical structure or properties beyond their unique identity. Each symbol serves as a label for the directed edges within the graph, dictating the specific transition that occurs from a given state based on the transition function $\delta(v,\sigma)$.

2. Symbol Encoding

Each symbol in the set Σ is encoded as a single byte, ranging from hexadecimal value 0x00 to 0xFF. This uniform encoding is applied consistently across all instances where symbols are used within the EchoPulse protocol, including the representation of the secret key (SK), the public key (PK), the random symbol sequence (r) used during encapsulation, and the labels of the edges defining the state transitions within G(V,E). Symbol sequences, such as SK, PK, and r, are stored and processed as ordered arrays of these byte-encoded symbols.

3. Symbol Interpretation

The symbols within Σ carry no intrinsic mathematical or logical semantics. Their meaning is solely defined by their role as inputs to the transition function $\delta(v,\sigma)$, which specifies the next state reached from a current state $v \in V$ upon processing the symbol $\sigma \in \Sigma$. This intentional lack of algebraic structure is a key design principle aimed at mitigating potential vulnerabilities related to algebraic side-channel attacks that might be applicable to cryptosystems based on mathematical groups or rings. The security of EchoPulse relies on the complex connectivity and dynamic evolution of the state transition graph, rather than any inherent properties of the symbols themselves.

4. Symbol Frequency & Distribution

For security and efficiency, sequences of symbols, such as the randomly generated sequence r used in encapsulation, should exhibit an approximately uniform distribution across the entire symbol set Σ. Any deterministic functions used in the derivation of keys or during the mutation process must be carefully designed to avoid introducing any statistical bias in the frequency of symbol occurrences. The mutation function μ should alter the graph structure and edge labels without skewing the overall distribution of symbols encountered during typical protocol operation.

5. Encoding Considerations

In implementing the EchoPulse protocol, symbols must be treated as raw byte values. No additional compression, padding, or reinterpretation of the symbol encoding should be performed. This direct byte-level handling simplifies implementation and reduces the potential for introducing subtle vulnerabilities. If EchoPulse is integrated into larger structured messages using serialization formats like CBOR (Concise Binary Object Representation) or ASN.1 (Abstract Syntax Notation One), the symbol sequences should be encoded as byte arrays according to the specifications of those formats, without any modification of the underlying symbol encoding.

6. Security Notes

The integrity and confidentiality of the symbol set Σ are important for the security of the EchoPulse protocol. No subset or partial information about the symbol set should be exposed or leaked to potential adversaries. Furthermore, implementations must be carefully scrutinized to prevent side-channel attacks that could potentially allow an attacker to infer the mapping between symbols and state transitions (i.e., the behavior of $\delta(v,\sigma)$). The dynamic mutation of the state transition graph over time enhances the unpredictability of the $\sigma \rightarrow v$ mappings, making it more challenging for an attacker to build a consistent model of the system's behavior across multiple sessions.

7. Conclusion

The symbol set Σ forms the elementary alphabet upon which the EchoPulse protocol operates. Defined as 256 abstract, non-algebraic entities, each symbol is encoded as a single byte and serves as a label for state transitions within the graph G(V,E). Their interpretation is solely within the context of the transition function δ. Maintaining a uniform distribution of symbols in generated sequences and ensuring their secure handling in implementation are crucial for the overall security of the EchoPulse KEM.

Version 1.0 — Symbol Table Layer — EchoPulse Initiative