DOCUMENT ID: EN-SPEC-V3.5.1

SUBJECT: ECHONODE GHOST-KEY STEGANOGRAPHIC PROTOCOL

CLASSIFICATION: TECHNICAL SPECIFICATION / INTERNAL USE ONLY

1. OVERVIEW

The EchoNode Ghost-Key Protocol is a high-entropy steganographic engine designed for the secure transmission of telemetry data. By masquerading as a Kernel-Mode Exception Dump (BSOD), it bypasses standard packet inspection and automated log filters. The protocol relies on a multi-layer approach where data is not merely encrypted, but hidden within the structural noise of a simulated software crash.

---

2. ARCHITECTURAL LAYERS

LAYER 1: CHARACTER MATRIX MAPPING

All plaintext is processed through an $8 \times 8$ Grid-Coordinate Cipher.

- **Mechanism:** Converts standard ASCII into a base-8 coordinate system.
- **Security:** Eliminates character-frequency signatures common in natural language.
- **Format:** Data is converted into a Morse-Binary stream (Dots/Dashes) representing the coordinates.

LAYER 2: THE ANCHOR & DYNAMIC KEY

The decoder requires two specific markers to initialize reconstruction:

- **The Signature Anchor:** !#BSOD#! — A hard-coded pointer that identifies the start of the data packet.
- **The Dynamic Skip-Key:** A 10-digit random integer immediately following the Anchor. This key defines the "skip" intervals for the specific session.

LAYER 3: SYNCHRONIZED SKIP-COUNTING

The core security layer. The message is fragmented and distributed across a stream of 43-character RNG noise.

- **The Logic:** The program reads the Skip-Key. If the key digit is $n$, the program ignores $n$ characters of noise and interprets the $n+1$ character.
- **Encoding Rule:** * **Numerics (0-9):** Represent a Morse "Dot."
    - **Alphanumerics (A-Z, Symbols):** Represent a Morse "Dash."

- **Result:** The data bit type is hidden in plain sight, as both types appear regularly in standard memory dumps.

LAYER 4: TERMINATOR & TAILING ENTROPY

To prevent file-size correlation (detecting message length by file volume), the protocol uses:

- **The Terminator:** %1O0X! — A unique hex-style marker signaling the end of the extraction logic.
- **RNG Flood:** Post-terminator, the system injects a randomized volume (0–1000 characters) of trailing noise. This ensures that every generated log file has a unique and unpredictable length.

---

3. AESTHETIC WRAPPING (CAMOUFLAGE)

The final payload is encapsulated in a **Virtual Studio Technology (VST)** debug template.

- **Header:** Randomly selects from a pool of audio-engine modules (e.g., Oscillator.vst3, ReverbUnit.dll).
- **Timestamp:** Real-time UTC injection for chronological authenticity.
- **Formatting:** 55-character line breaks to simulate a standard Hex-dump display.
- **Footer:** Uses a DUMP_COMPLETE status marker to provide a logical end-point for IT auditors.

---

4. OPERATIONAL SECURITY (OPSEC)

To maintain the integrity of the Ghost-Key protocol, users must:

1. **Rotate Anchors:** Periodically update the CORE_SIGNATURE to prevent pattern-matching.
2. **Contextual Use:** Only upload logs to repositories or threads where software debugging is expected (e.g., GitHub Issues, Dev-Forums).
3. **Key Secrecy:** Ensure the 10-digit skip logic remains confidential, as possession of the key and the anchor allows for instant reconstruction.

END OF SPECIFICATION