H.E.S. System Architecture

A Comprehensive, Resilient, and Transparent Election System Architecture

Objective: To detail the software, components, and infrastructure required for the end-to-end H.E.S. process, from voter registration to final result dissemination, ensuring maximum integrity and public trust.

By: Ed Millana 01-20-2026

Part 1: The H.E.S. System Architecture Diagram (Textual)

This diagram is structured in three primary layers that manage the entire election lifecycle.

```
================================================================
[      ELECTION MANAGEMENT SYSTEM (EMS) - "The Core"       ]
================================================================
|
|--- [ Voter Registration & Management Engine ]
|    |--- Secure Voter Database (Encrypted @ Rest)
|    |--- Biometric/PII Verification Module
|    |--- Polling Station Assignment Logic
|
|--- [ Candidate & Party Management Engine ]
|    |--- Candidate Qualification & Verification Module
|    |--- National & Local Database Integration
|    |--- Party List Management Module
|
|--- [ Ballot Design & Generation Engine ]
|    |--- Dynamic Ballot Layout Engine (DBLE)
|    |--- Software Localization & Translation Plugin (i18n/l10n)
|    |--- Accessibility & Compliance Module (WCAG)
|    |--- Secure Ballot Generation & Serialization Module (SBGSM)
|    |--- Integrated Printing Driver (for secure printers)
|
================================================================
[     IN-PRECINCT SYSTEM (IPS) - "The Frontline"        ]
================================================================
|
|--- [ Local Precinct Application (LPA) - Runs on a hardened laptop ]
|    |
|    |--- [ Scanner/Camera Interface Plugin ]
|    |    |--- Standard Webcam/Scanner Driver API
|    |
|    |--- [ Core Vision & Interpretation Engine ]
|    |    |--- Digital Character Recognition Engine
|    |    |--- Machine Learning Correction Model (e.g., for stray marks)
|    |    |--- Barcode/QR Code Reader (for ballot serial)
|    |
|    |--- [ User Interface (UI) Manager ]
|    |    |--- (Mode 1: Private Verification) -> Private Display Driver
|    |    |--- (Mode 2: Public Counting)     -> Public Display Driver
|    |
|    |--- [ Ballot Validation & Adjudication Module ]
|    |    |--- Over-vote/Under-vote Detection Logic
|    |    |--- Human Adjudication Interface
|    |    |--- Audit Log Engine (records all actions/changes)
|    |
|    |--- [ Local Data & Result Engine ]
```

```
|    |    |--- Encrypted Local Ballot Cache
|    |    |--- Result Aggregation Module (for precinct totals)
|    |    |--- VVPAT (Voter-Verified Paper Audit Trail) Generator
|    |    |--- Official Return (ER) Generation & Signing Module
|
|--- [ Hardware Interface Layer ]
|    |--- Voter's Private Screen (e.g., via HDMI to a small, angled monitor)
|    |--- Public Auditor's Screen (e.g., via HDMI to a large TV/LCD)
|    |--- Webcam & Sealed Ballot Box
|
================================================================
[  AGGREGATION & DISSEMINATION SYSTEM (ADS) - "The Network"  ]
================================================================
|
|--- [ Distributed Mesh Network Core (P2P DLT) ]
|    |--- Node Client (installed on each precinct laptop)
|    |    |--- Cryptographic Key Management (Public/Private Keys)
|    |    |--- Consensus Engine (e.g., Proof-of-Authority)
|    |    |--- Gossip Protocol (for data propagation)
|    |    |--- Immutable Ledger Storage (on each node)
|
|--- [ Public-Facing Infrastructure ]
|    |--- [ Real-Time Public Website Portal ]
|    |    |--- Web Framework Backend (e.g., Node.js, Python/Django)
|    |    |--- Data Aggregation API (connects to any mesh node)
|    |    |--- Frontend Framework (e.g., React, Vue.js)
|    |    |--- Interactive Map Component (e.g., Leaflet, Mapbox)
|    |    |--- Data Visualization Engine (e.g., D3.js, Chart.js)
|
|    |--- [ Security & Integrity Engine (Cross-Cutting) ]
|    |    |--- End-to-End Encryption (E2EE) Module (AES-256)
|    |    |--- Digital Signature Module (ECDSA)
|    |    |--- GNSS / GPS Timestamping Module
|    |    |--- Device Fingerprinting (IP/MAC Address) Logger
|
```

Part 2: Detailed Component Breakdown

A. Election Management System (EMS)

This is the central, high-security system used by the national/regional election commission before election day.

- **Voter Registration & Management Engine:**
    - **Software:** A custom-built enterprise application or a highly secured COTS product like a customized CRM.
    - **Components:**
        - **Secure Voter Database:** A PostgreSQL or Oracle database with Transparent Data Encryption (TDE). All Personally Identifiable Information (PII) is encrypted at rest and in transit. Access is strictly role-based.
        - **Biometric/PII Verification Module:** Plugins to interface with fingerprint, iris, or facial recognition hardware for de-duplication and verification.
        - **Polling Station Assignment Logic:** Algorithms to assign voters to specific precincts based on their registered address.

- **Candidate & Party Management Engine:**

- **Software:** A module within the EMS.
- **Components:**
    - **Candidate Qualification Module:** A workflow system that manages submission of requirements (e.g., certificates of candidacy, residency proofs).
    - **Database Integration:** Links to national government agencies (e.g., court records for disqualifications, COMELEC/SEC for party legitimacy).

- **Ballot Design & Generation Engine:**
    - **Software:** A specialized WYSIWYG (What You See Is What You Get) application, similar to a high-end desktop publisher but with election-specific logic.
    - **Components:**
        - **Dynamic Ballot Layout Engine (DBLE):** Automatically positions candidate names, party logos, and referendum text based on the number of contestants per race. It prevents layout errors.
        - **Software Localization Plugin (i18n/l10n):** Manages translations for all ballot text, including candidate names in different scripts, ensuring compliance with local language laws.
        - **Accessibility & Compliance Module:** Allows for the generation of ballots with large fonts, high contrast, or braille overlays for audio-tactile interfaces. Ensures WCAG compliance.
        - **Secure Ballot Generation & Serialization Module (SBGSM):** This is a critical security component.
            - It works with the DBLE to generate a unique, non-sequential, cryptographically random serial number for every single ballot.
            - It creates a secure, encrypted log linking a batch of serial numbers to a specific precinct and polling station. This log is never paired with a voter's identity.
        - **Integrated Printing Driver:** Interfaces directly with specialized, high-security printers that can embed microprint, watermarks, and heat-sensitive ink to prevent counterfeiting.

B. In-Precinct System (IPS)

This is the software running on the hardened laptop at each polling station.

- **Local Precinct Application (LPA):**
    - **OS:** A hardened Linux distribution (custom build with a minimal GUI) with Secure Boot enabled and full-disk encryption (LUKS).
    - **Core Vision & Interpretation Engine:**
        - **Character Recognation Engine:** The heart of the system. Open-source libraries like **MV** can be used, but it must be heavily trained and tested on sample ballots to achieve >99.9% accuracy.
        - **ML Correction Model:** A lightweight machine learning model trained to distinguish between valid votes and accidental smudges or coffee stains.
    - **Ballot Validation & Adjudication Module:**
        - **Human Adjudication Interface:** A simple, powerful UI that shows a high-resolution scan of the ambiguous ballot and allows multiple election officers to tag choices simultaneously. The final decision requires a digital signature or consensus PIN from the

officers and is logged immutably.

- **Result Engine:**
    - **ER Generation & Signing Module:** After all ballots are scanned and adjudicated, this module aggregates the precinct results. The final ER file is then cryptographically signed using the precinct's unique private key, creating a verifiable digital fingerprint.

C. Aggregation & Dissemination System (ADS)

This layer ensures results are securely and transparently transmitted and published.

- **Distributed Mesh Network Core:**
    - **Technology:** This is a private, permissioned blockchain or Distributed Ledger Technology (DLT). It's not a public, energy-intensive cryptocurrency blockchain.
    - **Node Client:** A lightweight application installed on every precinct laptop. It only communicates with other nodes (other precincts or authorized regional data centers).
    - **Consensus Engine (Proof-of-Authority):** Only pre-approved, authenticated nodes (the precinct laptops) are allowed to validate and add new "result blocks" to the chain. This is fast and efficient.
    - **Gossip Protocol:** When a precinct node broadcasts its signed ER, it sends it to its direct peers. Those peers validate the signature and then gossip it to their peers, and so on. This rapidly and resiliently propagates the data without a central point of failure.
- **Public-Facing Infrastructure:**
    - **Website Portal:**
        - **Data Aggregation API:** This is the magic. The website's backend doesn't connect to a single database. It can connect to *any* active node in the mesh network, query the ledger, and get a consistent, verified state of the election results.
        - **Interactive Map Component:** Pulls the GNSS location data, IP/MAC addresses, and other metadata directly from the signed transactions on the ledger, displaying it in an intuitive format.

Part 3: TO-DO: Suggestions and Critical Improvements

The proposed system is robust, but these areas require further strengthening for a real-world deployment.

1. **Enhanced Physical & Operational Security:**
    - **Supply Chain Security:** You must ensure the laptops, webcams, and other hardware are not tampered with *before* they arrive at the precinct. This requires a robust hardware attestation process.
    - **Laptop Hardening:** The OS should run from a read-only medium (like a live USB with a persistent, encrypted partition for the day's data). All non-essential ports (USB-A, etc.) should be physically disabled or epoxied shut.
    - **Two-Person Integrity (TPI):** All critical steps (e.g., starting the LPA, transmitting results, printing the ER) should require the presence and digital confirmation of two election officials from opposing parties.

2. **Voter & Poll Worker Training & Experience:**
    - **Simulated Software:** A training version of the LPA with simulated ballots is essential for poll workers to practice adjudication and error-handling before election day.
    - **Voter Instructions:** The private verification screen needs to be incredibly simple and clear. Use icons, large fonts, and minimal text to reduce voter confusion. A short, mandatory instructional video could play on the main screen before a voter approaches.

3. **Resilience & Redundancy:**
    - **Power Outages:** Each precinct must have a certified UPS (Uninterruptible Power Supply) capable of running the laptop, screens, and webcam for at least 6-8 hours. The LPA software featured an instant "hibernate" and "resume" state that does not lose data.
    - **Network Isolation:** The mesh network should be tested for scenarios where precincts are geographically isolated. The system must function perfectly even if a node can only connect to one other node before syncing up with the wider network later.
    - **Component Failure:** What happens if the webcam fails? A procedure is needed involving a designated contingency precinct scanner or manual counting with strict oversight.

4. **Addressing Coercion and Vote Selling:**
    - While the private screen helps, a determined bad actor could still force a voter to show them the screen. An improvement could be a **"Vote Coercion Resistance" feature**: on the private verification screen, alongside the correctly interpreted choices, there could be a button labeled "Confirm & Lock Ballot" and a seemingly innocent but different button (e.g., styled as a "Language" or "Help" icon). Pressing the "innocent" button also locks the ballot but secretly registers a "coercion alert" with poll workers without alerting the coercer. This is a complex but critical feature for high-risk areas.

5. **Post-Election Auditing:**
    - The system is designed for transparency, but a formal, legally mandated Risk-Limiting Audit (RLA) must be baked into the process. The software should have a dedicated "Audit Mode" that randomly selects a statistically significant number of paper ballots from the sealed box for a manual hand count, which is then compared to the digital record from the mesh network.

This is a Preliminary document. All information is subject to change without notice due to the system's active development.

Prepaired by: Ed Millana – 01-20-2026

Email: ed@millawave.com

🌐 Learn More:

🔗 www.hybridelection.net

Follow HES on social media for demos and pilot results