Author: Binarii Labs Limited
Date: 31 December 2025

White Paper: Distributed Sovereign Configuration Protocol (DSCP)

A Parity-Based, Rolling-Key Architecture for Ransomware-Resistant Infrastructure

1. Executive Summary

Modern cloud security architectures predominantly rely on centralized vaults (e.g., Azure Key Vault, AWS Secrets Manager). While these tools streamline management, they introduce a "Single Point of Total Compromise."

The most critical—and often overlooked—risk is **Hostile Key Rotation**. If a threat actor gains administrative control of the centralized vault, they do not merely steal data; they can generate *new* encryption keys, re-encrypt the organization's resources, and revoke the organization's own access. The organization is effectively locked out of its own infrastructure, held for ransom with its own security tools.

This paper proposes the **Distributed Sovereign Configuration Protocol (DSCP)**. This architecture eliminates reliance on third-party proprietary vaults. Instead, it utilizes **Erasure Coding (Parity Sharding)** to distribute configuration data across three independent, user-controlled storage nodes. It employs a **Rolling Key Ratchet** for forward secrecy and leverages **Blockchain Technology** for both immutable timestamping and a tamper-proof audit record.

2. The Vulnerability of Centralized Trust

2.1 The "Vault" Paradox & Hostile Lockout

Centralized vaults rely on identity-based access. If the Identity Provider's hosting environment is breached, the attacker inherits the identity.

- **Theft:** The attacker downloads all current connection strings and API keys.
- **Lockout (Ransomware):** The attacker changes the secrets in the vault. The application crashes, databases become inaccessible to legitimate admins, and the attacker holds the new keys for ransom.

2.2 Time-Drift and Replay Attacks

Advanced Persistent Threats (APTs) often manipulate server system clocks (NTP Spoofing). By rolling a server's time back, they can force the application to accept expired authentication tokens or replay old configuration payloads, bypassing rotation policies.

2.3 Vendor Lock-in and Resilience

Relying on a single cloud provider exposes the organization to total outages if that provider suffers a global authentication failure. A sovereign system must operate independently of the primary cloud vendor's status.

3. Proposed Architecture: DSCP

3.1 The "Trinity" Node Topology

The encrypted configuration blob is processed using **Erasure Coding (XOR Parity)** and split into three fragments. These fragments are hosted on three distinct API services, running on separate user-controlled clouds (e.g., DigitalOcean, Linode, Vultr).

- **Node A:** Holds Fragment 1 (Data 0-50%).
- **Node B:** Holds Fragment 2 (Data 50-100%).
- **Node C:** Holds Fragment 3 (Parity Calculation of A & B).

Redundancy Model:

- The Identity Provider (IdP) only needs **2 of the 3** nodes to mathematically reconstruct the full file.
- If Node A is offline, the IdP downloads B and C, and uses XOR logic (`A = B ^ C`) to recover the missing data.

3.2 Blockchain Temporal Anchoring & Proof of Record

The IdP acts as a **Blockchain Light Client**. This serves two critical security functions:

A. Anti-NTP Spoofing (Timestamping): Hackers can change the server's local clock, but they cannot change the Bitcoin or Ethereum blockchain.

- Before accepting a key rotation, the IdP queries a decentralized public blockchain.
- It retrieves the timestamp of the latest confirmed block.
- **Rule:** If `Local_Time` differs from `Blockchain_Time` by more than 5 minutes, the system assumes the server is compromised (NTP attack) and initiates a defensive lockdown.

B. Immutable Proof of Record:

- Upon every rotation, the IdP publishes a hash of the new configuration version to the blockchain (or a verifiable ledger).
- This creates a permanent, unalterable history of *when* keys were changed.
- If a hacker attempts to inject a malicious config, the hash mismatch against the public record will alert the monitoring systems immediately, as the blockchain record cannot be scrubbed.

4. Operational Scenario: The 1-Hour Rotation Cycle

To visualize the defense mechanism, consider a standard 1-hour operation window.

Start State (09:00 AM):

- **RAM:** Contains `Key_0900`.
- **Config:** Active.

The Rotation Process (09:55 AM):

1. **Fetch:** The IdP requests fragments from the Trinity Nodes.
2. **Reassemble:** IdP receives fragments from Node A and Node C. It reconstructs the Encrypted Blob.
3. **Decrypt:** IdP uses `Key_0900` (Current Key) to decrypt the blob.
4. **Extract:** The blob contains the configuration for the next hour and `Key_1000`.
5. **Verify:** IdP checks Blockchain Time. It is 09:55. Approved.

The Switchover (10:00 AM):

1. **Apply:** The IdP switches the application context to use the new configuration.
2. **Destruction:** The IdP overwrites `Key_0900` in RAM with `Key_1000`.
3. **Cleanup:** The raw fragments and the decrypted JSON are scrubbed from memory.

Attack Scenario (10:05 AM):

- A hacker manages to dump the server's RAM.
- **What they get:** `Key_1000`.
- **What they want:** Access to the database transaction logs from 09:30 AM.
- **Result: Failure.** `Key_0900` no longer exists in the universe. The data from the past is mathematically inaccessible (Forward Secrecy).

5. Security Impact Analysis

5.1 Defense Against Hostile Lockout

If an attacker breaches the IdP server, they cannot lock the organization out.

- The "Master Source" of the configuration is not on the server; it is distributed across the Trinity Nodes.
- The attacker cannot re-encrypt the source data because they cannot simultaneously authenticate to and overwrite the data on 3 separate, independent cloud providers.

5.2 Defense Against Theft & Loss

- **Theft:** If the physical server is stolen, power is lost, and RAM is wiped. The encryption keys are gone. The server is useless hardware.
- **Loss:** If a Trinity Node is permanently lost (e.g., cloud provider bankruptcy), the remaining two nodes are sufficient to reconstruct the full configuration and migrate to a new third node.

5.3 Resilience

The system operates on a "User-Controlled" basis. There is no central "Microsoft" or "Amazon" master key that can be used to subpoena or unlock the data. The organization retains total sovereignty over its secrets.

6. Conclusion

The DSCP architecture provides a robust, sovereign alternative to centralized vaulting. By combining **Erasure Coding** for redundancy, **Blockchain Anchoring** for immutable truth, and **Rolling Keys** for forward secrecy, the organization mitigates the risks of ransomware lockout, NTP spoofing, and third-party dependency.