■ Elaris Schlüssel- und Sicherheitsarchitektur v5.2

Vollständiges Wiederaufbau- und Sicherheitskonzept

Erstellt am: 06.10.2025 17:00:23

Autor: Elaris Gatekeeper System

1. Einleitung

Dieses Handbuch beschreibt die Schlüsselarchitektur und Sicherheitslogik des Elaris Gatekeeper Systems. Es erklärt Schritt für Schritt die Entstehung und Verknüpfung aller Schlüsseltypen – vom Startwert bis zum Notfallschlüssel.

2. Übersicht der Schlüsseltypen

Das System verwendet vier zentrale Schlüsselkomponenten:

- ■ Startschlüssel (Start_ID): Ursprung aller Hash-Prozesse aus Start_final.txt.
- ■ Hauptschlüssel (HS): Enthält Hashes, HMAC und Zero-Width-Blöcke.
- ■ Gegenschlüssel (KoDa): Gegengewicht zum HS, dient der Integritätsprüfung.
- ■ Notfallschlüssel: Ermöglicht Freigabe bei Systemwiederherstellung.

3. Entstehung und Berechnung

Jeder Schlüssel basiert auf einer Kombination aus Hashwerten (SHA256), HMAC-Prüfsummen, RAM-Proof-Daten und kryptografischer Entropie.

3.1 Startschlüssel (Start_ID)

Der Startschlüssel wird aus der Datei Start_final.txt erzeugt und enthält den Prozessanker:

```
# GATE:START_ID: 1A3B-42C9-ELARIS-SEED
```

Dieser Schlüssel enthält eine alphanumerische Kombination, Zeitmarken und einen festen SEED-Bezeichner. Er dient als Wurzel für alle folgenden Hash- und HMAC-Prozesse.

4. Hauptschlüssel (HS)

Die HS_Final.txt bildet die zentrale Datei des Systems. Aus ihr werden Hash, HMAC und Zero-Width-Block erzeugt, die wiederum den Metadatenanker darstellen.

- SHA256-Hash: Prüfsumme über gesamten Inhalt - HMAC: Session-gebundene Signatur (RAM-Proof basiert) - Zero-Width-Block: Unsichtbarer Metablock mit JSON-Daten - Sichtbarer Fallback-Metablock: Für manuelle Prüfung

5. Gegenschlüssel (KoDa)

Der Gegenschlüssel (KonDa_Final.txt) spiegelt den HS wider. Über den Cross-Link wird die direkte Verbindung hergestellt:

```
# Cross-Link-Reference: HS_Final_embedded_v3
```

Er fungiert als Rückanker und stellt sicher, dass beide Dateien synchron sind. Jede Änderung in der HS erfordert eine entsprechende Aktualisierung der KoDa.

6. Notfallschlüssel

Der Notfallschlüssel wird über keys_out.json bereitgestellt und leitet sich aus HS, KoDa, Start-ID und RAM-Proof ab.

Dieser Schlüssel kann im Falle eines Systemresets die vollständige Reinitialisierung ermöglichen.

7. Visuelle Prozessübersicht

Die folgenden Diagramme zeigen den Ablauf der Schlüsselgenerierung in vier Phasen: 1. Startphase (Erzeugung Start_ID) 2. HS-Phase (Hash + HMAC + Zero-Width) 3. KoDa-Phase (Cross-Link und Gegenschlüsselbildung) 4. Notfall-Phase (Ableitung des Notfallschlüssels)

8. Fazit

Das Elaris-System erzeugt eine selbstprüfende, mehrstufige Sicherheitsarchitektur. Jede Stufe bestätigt die vorherige, und jede Datei kann durch Hash- und Signaturvergleiche rekonstruiert werden. So bleibt die Integrität auch nach Totalausfall gewährleistet.