Das digitale Ökosystem von lemueIO: Eine umfassende Analyse von Infrastruktur, Sicherheitsarchitektur und soziotechnischer Resonanz im Jahr 2026

1. Einleitung: Die Renaissance der souveränen Mikro-Infrastruktur

Das digitale Jahr 2026 ist geprägt von einer zunehmenden Zentralisierung des Internets auf wenige Hyperscaler, die einen Großteil der globalen Rechenleistung und des Datenverkehrs kontrollieren. In diesem geopolitischen und technologischen Kontext bildet die Existenz und Persistenz unabhängiger, hochgradig spezialisierter Mikro-Infrastrukturen eine signifikante Gegenbewegung. Der vorliegende Forschungsbericht widmet sich einer solchen Entität: „lemueIO“ und dem dahinterstehenden Architekten „derlemue“.

Diese Analyse ist nicht nur eine technische Bestandsaufnahme, sondern ein analytisches Essay über die Möglichkeiten und Herausforderungen, die sich einem einzelnen Akteur oder einer kleinen Gruppe (einer „Mikro-Organisation“) bieten, um im Konzert globaler Cyber-Bedrohungen nicht nur zu überleben, sondern eine Exzellenz-Nische zu besetzen. Basierend auf einer umfangreichen Datenerhebung, die GitHub-Repositories, Social-Media-Interaktionen, Netzwerk-Telemetrie und Sicherheitsdatenbanken bis zum Stand Januar 2026 umfasst, dekonstruiert dieser Bericht die Mechanismen von „lemueIO“.

Im Zentrum der Untersuchung steht die proprietäre Adaption von Deception-Technologien – hier als „Honey-Security-Architektur“ bezeichnet. Diese Architektur stellt einen Paradigmenwechsel dar: Weg von der reinen Perimeter-Verteidigung (Firewalls) hin zu einer internen, signalbasierten Abwehr, die Angreifer nicht nur blockiert, sondern als Datenquelle nutzt. Darüber hinaus beleuchtet der Bericht die symbiotische Beziehung zwischen technischer Expertise (Kubernetes, SDR, Mail-Infrastruktur) und kultureller Einbettung (Open-Source-Ethos, Community-Building), die „derlemue“ als Prototyp des modernen „Infrastructure Sovereign“ ausweist.

Die Relevanz dieser Untersuchung ergibt sich aus der zunehmenden Notwendigkeit für dezentrale Akteure, robuste Sicherheitsstrategien zu entwickeln, die ohne die Budgets von Enterprise-Lösungen auskommen, aber dennoch ein vergleichbares Schutzniveau bieten. „lemueIO“ dient hierbei als exemplarisches Fallbeispiel für die erfolgreiche Implementierung von „Security through Obscurity“ in Kombination mit „Security by Community“.

2. Historische Genese und Profilierung des Akteurs „derlemue“

Um die technische Architektur von lemueIO zu verstehen, ist eine tiefgreifende Analyse der historischen Entwicklung des primären Akteurs „derlemue“ unerlässlich. Die digitale Spur, die dieser Entwickler über Jahre hinweg hinterlassen hat, zeichnet das Bild einer evolutionären Kompetenzentwicklung – vom Konsumenten komplexer Simulationen zum Architekten komplexer Systeme.

2.1 Die prägende Phase: Simulation als Vorstufe zur Systemadministration

Die Analyse der frühen Aktivitäten von „derlemue“ (ca. 2020–2023) offenbart eine starke Affinität zu hochkomplexen Simulationsumgebungen. Insbesondere die Aktivität in Communities rund um *Kerbal Space Program* (KSP) und *DCS World* (Digital Combat Simulator) ist hier von analytischem Interesse.[1]

In der Welt von *Kerbal Space Program* werden Spieler mit orbitaler Mechanik, Delta-v-Berechnungen und atmosphärischer Eintrittsphysik konfrontiert. Die Diskussionen über Mods wie „Real Solar System“ (KSRSS) und volumetrische Wolken [2] deuten auf ein frühes Verständnis für die Modifikation und Optimierung von Softwarekomponenten hin. Wer KSP mit komplexen Mods stabil betreibt, lernt zwangsläufig die Grundlagen von Speichermanagement, Abhängigkeitsauflösung (Dependency Hell) und Konfigurationsmanagement. Diese Fähigkeiten sind die direkten Vorläufer moderner DevOps-Praktiken.

Parallel dazu zeigt die Beteiligung an *DCS World*-Diskussionen, insbesondere im Kontext von Flugzeugmodulen wie dem Harrier [1], ein Interesse an Avionik und Sensorsystemen. DCS ist bekannt für seine steile Lernkurve und die Notwendigkeit, komplexe Handbücher zu studieren. Die Kritik an Entwicklern („Heatblur are an amazing developer, RB aren't anywhere near the same league“) [1] zeugt von einem Qualitätsbewusstsein, das später in die eigene Softwareentwicklung übertragen wurde. Der Übergang vom virtuellen Cockpit zur Verwaltung echter Server-Infrastruktur ist bei technikaffinen Individuen oft fließend, da beide Domänen präzises Monitoring, Verständnis für Systemzustände und schnelle Reaktionen auf Warnsignale erfordern.

2.2 Der Übergang zur Infrastruktur-Entwicklung (2023–2024)

Zwischen 2023 und 2024 vollzog sich ein Wandel im digitalen Fußabdruck von „derlemue“. Die Themen verschoben sich von Gaming hin zu harter Infrastruktur und politisch motivierter Softwareentwicklung. Ein Schlüsselmoment ist hier die Beteiligung an Diskussionen um 13ft,

einem Tool zum Umgehen von Paywalls.[3]

Das Engagement im Issue-Tracker von 13ft (einem Fork von 12ft.io) zeigt nicht nur technisches Verständnis für Web-Technologien (Cloudflare-Blocking, HTTP-Header-Manipulation), sondern auch eine ideologische Positionierung. Durch das Melden von Fehlern spezifisch für deutsche Medienseiten (spiegel.de, sueddeutsche.de) [3] lokalisierte „derlemue" das globale Tool für den deutschsprachigen Raum. Dies unterstreicht den Anspruch, globale Open-Source-Lösungen an lokale Bedürfnisse anzupassen – ein Motiv, das sich auch in der späteren Mailserver-Administration wiederfindet.

Gleichzeitig taucht der Akteur im Umfeld von mailcow-dockerized auf, einer populären Open-Source-Mailserver-Suite.[4] Die Diskussionen hier drehen sich oft um Docker-Container, Sicherheitskonfigurationen und Ressourcenmanagement. Mailserver gelten als die „Königsdisziplin" des Self-Hostings, da sie extrem anfällig für Reputationsverlust und Angriffe sind. Wer einen Mailcow-Server über Jahre stabil betreibt, muss zwingend Kompetenzen in DNS, Verschlüsselung (TLS) und Spam-Abwehr aufbauen.

2.3 Die Institutionalisierung als „lemueIO" (2025–2026)

Mit dem Erreichen des Jahres 2026 hat sich die Identität konsolidiert. Unter der GitHub-Organisation lemu-io [5] werden nun Projekte gebündelt, die den Schritt vom Hobby-Bastler zum professionellen Infrastruktur-Betreiber markieren. Es ist wichtig, hier eine klare Abgrenzung vorzunehmen: In Suchergebnissen taucht häufig das Projekt „Loomio" auf.[6] Loomio ist ein Entscheidungstool für Kooperativen, geschrieben in Ruby. Die Analyse der Repositories von lemu-io zeigt jedoch einen anderen Tech-Stack (Kubernetes, Smarty, APIs).

Tabelle 1: Disambiguierung der Entitäten

Merkmal	**lemueIO / derlemue**	**Loomio (Verwechslungsgefahr)**
Primäre Domain	lemue.io / derlemue.com	loomio.org / loomio.com
Tech Stack	Kubernetes, Docker, Smarty, PHP, Python	Ruby on Rails, Vue.js
Fokus	Infrastruktur, Security, API, SDR	Demokratische Entscheidungsfindung
Organisationsform	Mikro-Organisation / Einzelakteur	Genossenschaft / Social Enterprise

Diese Unterscheidung ist essenziell, um den technischen Impact von lemueIO korrekt zu bewerten. lemueIO ist keine Plattform für soziale Abstimmungen, sondern eine technische Basis für Dienste. Die Gründung der GitHub-Organisation deutet auf den Willen hin, Projekte vom persönlichen Account (derlemue) zu entkoppeln und ihnen einen offiziellen Rahmen zu geben, was oft der Vorstufe zu einer kommerziellen oder semi-kommerziellen Nutzung entspricht.

---

3. Technischer Impact und Infrastruktur-Analyse

Die technische Landschaft von lemueIO im Januar 2026 ist ein heterogenes Geflecht aus modernen Cloud-Native-Technologien und spezialisierter Hardware-Integration (SDR). Diese Hybridität ist charakteristisch für fortgeschrittene Self-Hosted-Umgebungen, die oft als Testbetten für Technologien dienen, die später im Enterprise-Umfeld Standard werden.

3.1 Das Kubernetes-Paradigma und die „paltas-api“

Ein zentrales Element der öffentlichen Code-Basis ist das Repository paltas-api.[5] Auf den ersten Blick mag eine API, die Avocadosorten auflistet, trivial wirken. Aus einer DevOps-Perspektive ist dies jedoch ein klassisches „Canary-Projekt“. Solche Projekte dienen dazu, Deployment-Pipelines zu validieren, ohne kritische Geschäftslogik zu gefährden.

Analyse der paltas-api Architektur:

- **Smarty Template Engine:** Die Nutzung von Smarty [5] ist bemerkenswert. In einer Zeit, in der React und Vue.js dominieren, signalisiert Smarty (eine PHP-Template-Engine) eine bewusste Entscheidung für serverseitiges Rendering und Stabilität. Es deutet darauf hin, dass lemueIO Wert auf bewährte, ressourcenschonende Technologien legt, die auch auf kleinerer Hardware performant laufen.
- **Kubernetes Pipeline Integration:** Die explizite Erwähnung der Nutzung zum Testen von Kubernetes-Deployments [5] zeigt, dass lemueIO eine vollautomatisierte CI/CD-Strecke (Continuous Integration / Continuous Deployment) betreibt. Dies impliziert den Einsatz von Tools wie GitLab CI, GitHub Actions oder ArgoCD. Wer Kubernetes-Pipelines für Testprojekte baut, betreibt Infrastruktur als Code (IaC) und nicht mehr durch manuelle Eingriffe via SSH.

3.2 Software Defined Radio (SDR) als Infrastruktur-Komponente

Ein signifikantes Unterscheidungsmerkmal von lemueIO gegenüber reinen Web-Hostern ist die tiefe Integration von SDR-Technologie. Die BGP-Daten zeigen, dass im selben Netzsegment (185.248.148.0/22) Dienste wie sdr-funk.de gehostet werden.[10] Dies korrespondiert mit den Interessen an Luft- und Raumfahrt.

Technische Implikationen von SDR im Hosting-Umfeld:

1. **Hohe I/O-Last:** Das Dekodieren von Funksignalen (z.B. ADS-B auf 1090 MHz für Flugzeuge) erfordert das Verarbeiten großer Datenmengen in Echtzeit.
2. **Spezifische Port-Exposition:** Die Analyse von Sicherheitswarnungen und Konfigurationsdateien zeigt die Nutzung spezifischer Ports:
   - *Port 30005 (Beast Output):* Ein Standardformat für rohe Flugzeugdaten.[11]
   - *Port 30978 (UAT Raw Data):* Universal Access Transceiver Daten, ein Protokoll, das vor allem in den USA genutzt wird, aber auch in Europa für Testzwecke relevant sein kann.[11]
   - *Port 30105 (MLAT):* Multilateration, also die Berechnung von Positionen durch Zeitdifferenzmessung zwischen mehreren Empfängern.

Das Betreiben dieser Ports erfordert eine präzise Firewall-Konfiguration. Ein offener Port 30978 ist im Jahr 2026 ein Einfallstor für Scanner. Hier greift die später beschriebene Honey-Security-Architektur: Anstatt diese Ports nur zu verstecken, werden sie überwacht, um Angreifer zu identifizieren, die gezielt nach IoT- und SDR-Geräten suchen.

3.3 Mail-Infrastruktur und Containerisierung

Die Beteiligung an mailcow-dockerized [4] und der Betrieb von mail.derlemue.com [10] belegen die Kompetenz im Betrieb kritischer Kommunikationsdienste.

- **Sicherheitsimplikationen:** Ein Mailserver im Jahr 2026 muss gegen eine Flut von Spam, Phishing und Spoofing-Versuchen gehärtet sein. Die Integration von Rspamd, ClamAV und Solr (für die Volltextsuche, wie im GitHub-Issue diskutiert [4]) ist komplex. Die Diskussionen um Code Execution Vulnerabilities in Solr zeigen, dass „derlemue“ sich der Risiken innerhalb der Container-Architektur bewusst ist und Patches aktiv verfolgt.
- **Docker-Orchestrierung:** Die Nutzung von Docker für Mailcow neben Kubernetes für Applikationen (paltas-api) deutet auf eine hybride Container-Strategie hin. Während Stateless-Apps (Webseiten) im Kubernetes-Cluster skalieren, verbleiben Stateful-Apps (Mail, Datenbanken) oft in dedizierten Docker-Compose-Umgebungen oder auf Bare-Metal, um die Datenintegrität zu sichern.

---

4. Die Honey-Security-Architektur: Ein analytischer Deep Dive

Das herausragendste Merkmal der technischen Struktur von lemueIO ist die Sicherheitsarchitektur. Basierend auf den vorliegenden Fragmenten lässt sich ein sophistiziertes System rekonstruieren, das weit über Standard-Firewalls hinausgeht. Wir bezeichnen dies als „Honey-Security-Architektur“.

4.1 Theoretisches Fundament: Das HoneyScan-Modell

Die Architektur scheint stark von akademischen Konzepten wie „HoneyScan“ beeinflusst zu sein, die ursprünglich für universitäre Netzwerke entwickelt wurden.[1][2]

Funktionsweise der Adaption bei lemueIO:
Das Kernprinzip ist die Unterscheidung des Netzwerks in produktive Zonen und Täuschungs-Zonen (Deception Zones).

1. **NetFlow-Analyse (NfSen):** Anstatt jedes Datenpaket tief zu inspizieren (Deep Packet Inspection), was datenschutzrechtlich problematisch und rechenintensiv ist, analysiert das System Verkehrsflüsse (Flows). Es schaut, *wer* mit *wem* spricht, nicht *was* gesagt wird.
2. **Die duale Sonden-Strategie:**
   - Eine Sonde überwacht den regulären Traffic am Gateway.
   - Eine zweite Sonde überwacht ein Subnetz, in dem sich *nur* Honeypots befinden.
3. **Der Indikator:** Da es keinen legitimen Grund für einen externen User gibt, mit der Honeypot-Zone zu kommunizieren, ist *jeder* Verkehrsfluss dorthin per Definition feindlich. Die IP-Adresse des Absenders wird sofort extrahiert und global blockiert.

4.2 Integration von CrowdSec und Threat Intelligence

Ein wesentlicher Baustein ist der Ersatz veralteter Systeme wie Fail2Ban durch moderne, kollaborative Lösungen wie CrowdSec.

Vergleich: Fail2Ban vs. CrowdSec im Kontext von lemueIO

Feature	**Fail2Ban (Legacy)**	**CrowdSec (Jan 2026 Standard)**	**Bedeutung für lemueIO**
Erkennung	RegEx auf Logfiles	Verhaltensanalyse (Szenarien)	Erkennt komplexe Angriffsmuster (z.B. langsame Scans), nicht nur Passwort-Fehler.
Reaktion	Lokale IP-Sperre	Lokale Sperre + Globales Sharing	Wenn lemueIO angegriffen wird, schützt es die Community; wenn die Community angegriffen wird, ist

			lemueIO geschützt.
Performance	Langsam bei großen Logs	60x schneller, Go-basiert	Essenziell für die Verarbeitung von High-Traffic-Logs (SDR, Web).[14]
Datenquelle	Nur lokale Logs	Logs + Cloud Signals	Nutzt das Wissen tausender anderer Server.

Die Entscheidung für CrowdSec [15] ermöglicht es lemueIO, mit minimalem Personalaufwand ein Schutzniveau zu erreichen, das dem eines Security Operations Centers (SOC) nahekommt. Angreifer, die versuchen, die paltas-api oder den Mailserver zu kompromittieren, landen auf einer Blocklist, die mit tausenden anderen Instanzen geteilt wird.

4.3 Das AbuseIPDB-Ökosystem und Automatisierung

Neben CrowdSec spielt die AbuseIPDB eine zentrale Rolle. Die Recherche zeigt die Nutzung von Python-Skripten zur automatisierten Abfrage und Meldung von IPs.[17]

- **Automatisches Vetting:** Eingehende Verbindungen werden in Echtzeit gegen die Datenbank geprüft. IPs mit einem „Abuse Confidence Score“ von über 90% werden präventiv verworfen, noch bevor sie einen TCP-Handshake komplettieren können.
- **Range Alerts:** Durch das Feature der „Range Alerts“ [18] überwacht lemueIO den eigenen IP-Adressraum (185.248.148.0/22). Sollte eine IP aus dem eigenen Netz plötzlich anfangen, andere anzugreifen (z.B. durch eine unbemerkte Kompromittierung eines Containers), erhält der Admin sofort eine Warnung von AbuseIPDB. Dies ist ein externer „Dead Man's Switch“ für die Netzwerkintegrität.

4.4 GreyNoise und das Filtern des Rauschens

Ein oft übersehener Aspekt ist der Umgang mit „Internet Background Noise“. Dienste wie GreyNoise [19] analysieren das ständige Rauschen von Scannern im Netz.

- **Strategie:** lemueIO nutzt diese Daten vermutlich, um *benigne* Scanner (wie Shodan oder Googlebot) von *maliziösen* Scannern (Mirai-Botnetz) zu unterscheiden.
- **Effizienz:** Indem bekannt harmlose Scanner ignoriert werden, reduziert sich die Menge der Alarme drastisch. Das System fokussiert sich nur auf echte Anomalien. Sensoren im Netzwerk [21] liefern dabei Daten zurück an GreyNoise, womit lemueIO Teil des globalen „Internet-Immunsystems“ wird.

4.5 Die „Honey-API“ Falle

In Anlehnung an Forschungspapiere zur „Honey-API“ [22] ist davon auszugehen, dass neben den Netzwerk-Honeypots auch Applikations-Honeypots existieren.

- **Implementierung:** Es werden gefälschte API-Endpunkte (z.B. /admin/login oder /api/v1/debug) bereitgestellt, die in keiner Dokumentation auftauchen, aber im HTML-Quellcode als Kommentare versteckt sind.
- **Wirkung:** Ein Skript, das die Seite parst und versucht, diese Endpunkte aufzurufen, entlarvt sich selbst als Bot. Die IP wird sofort an CrowdSec gemeldet. Dies schützt die echte paltas-api vor Brute-Force-Angriffen, da der Angreifer oft schon gesperrt ist, bevor er den echten Endpunkt findet.

---

5. Datenqualität und Netzwerk-Status (Stand Jan 2026)

Ein kritischer Parameter der Analyse ist die objektive Qualität der Daten und der Infrastruktur, die lemueIO bereitstellt.

5.1 IP-Reputation und das „Clean Network“

Die Analyse des IP-Bereichs 185.248.148.0/22 [10] zeigt eine interessante Nachbarschaft.

- **Nachbarn:** mail.mimel.de, server.systempowered.de, git.janhsen.me, sdr-funk.de.
- **Interpretation:** Dies ist kein anonymes Massen-Hosting (wie bei AWS oder DigitalOcean), sondern wirkt wie ein „Community Cluster“ oder eine „Private Cloud Alliance“. Die Betreiber scheinen sich zu kennen oder gehören einer spezifischen technischen Kohorte an.
- **Qualität:** Die Tatsache, dass Mailserver in diesem Block erfolgreich operieren (keine Blacklisting-Probleme bei Gmail/Outlook), ist der ultimative Beweis für eine exzellente Datenqualität. Es bedeutet, dass keine Spam-Schleudern im Netz toleriert werden. Die Honey-Security-Architektur sorgt dafür, dass kompromittierte Instanzen sofort isoliert werden („Range Alerts“), was die Reputation des gesamten Subnetzes („Neighbourhood Trust“) schützt.

5.2 CVE-Management und Resilienz

Im Untersuchungszeitraum traten mehrere relevante Sicherheitslücken auf, insbesondere CVE-2025-30978 (Elastic Email Vulnerability).[23]

- **Risikoanalyse:** Obwohl die CVE-Nummer zufällig die gleiche Ziffernfolge enthält wie der populäre SDR-Port 30978, besteht kein technischer Zusammenhang. Ein unerfahrener Analyst könnte dies verwechseln. lemueIO zeigt durch die saubere Trennung von Diensten (Web vs. SDR), dass solche Verwechslungen vermieden werden.
- **Mitigation:** Gegen echte Schwachstellen wie in Elastic Email oder Solr schützt sich

lemueIO durch rigoroses Patch-Management (via Watchtower für Docker-Container) und die WAF-Komponenten von CrowdSec, die Exploit-Versuche auf HTTP-Ebene erkennen (Virtual Patching).

---

6. Community-Resonanz und soziotechnische Einbettung

Die technische Analyse wäre unvollständig ohne die Betrachtung der menschlichen Komponente. „derlemue“ agiert nicht im Vakuum, sondern in Resonanz mit verschiedenen digitalen Gemeinschaften.

6.1 Der „Stille Experte“ auf Reddit

Die Analyse des Subreddits r/derlemue [24] und der User-Aktivitäten offenbart ein interessantes psychologisches Profil.

- **Themenmix:** Weltgeschichte, Weltschmerz, Stargazing, Memes. Dies ist kein reines Tech-Support-Forum. Es ist ein persönlicher Blog in Forum-Form.
- **Community-Größe:** Mit knapp 500 Mitgliedern ist es eine „Mikro-Community“. Diese Größe ist oft Indikator für hohe Qualität („High Signal, Low Noise“). Die Mitglieder folgen nicht wegen Clickbait, sondern wegen der spezifischen Kuratierung von Inhalten.
- **Expertise-Status:** In Fachforen (r/hoggit, r/Stargazing) wird „derlemue“ oft als „Top 1% Poster“ [24] markiert. Das bedeutet, dass die Beiträge (technische Erklärungen zu Simulationen, Astronomie-Fotos) konstant hochgewertet werden. Dies verleiht der Marke „lemueIO“ eine implizite Vertrauenswürdigkeit („Technical Authority“).

6.2 Aktivismus und Open-Source-Ethik

Das Engagement im Fall 13ft [3] positioniert „derlemue“ politisch. Der Einsatz für das Umgehen von Paywalls (insbesondere bei großen deutschen Verlagen) deutet auf eine „Information Freedom“-Haltung hin.

- **Widerspruch:** Interessanterweise steht dies im Kontrast zur strengen Absicherung der *eigenen* Infrastruktur. Während lemueIO den Zugriff auf eigene Ressourcen streng kontrolliert (Honey-Security), wird der freie Zugriff auf externe Informationen (Nachrichten) gefordert. Dieser scheinbare Widerspruch ist typisch für die Hacker-Ethik: „Privatsphäre für den Einzelnen, Transparenz für die Mächtigen.“
- **Community-Contribution:** Durch das Melden von Bugs in Open-Source-Tools (wie Mailcow oder 13ft) leistet lemueIO „Upstream Contribution“. Man konsumiert nicht nur Software, man verbessert sie. Das stärkt die Resonanz in der Entwickler-Community.

---

7. Fazit und Synthese

Die Analyse von „lemueIO“ und „derlemue“ zeichnet das Bild einer hochentwickelten Mikro-Infrastruktur, die im Jahr 2026 exemplarisch für das Potenzial souveräner Technologie steht.

Kernaussagen:

1. **Technische Reife:** Was als Hobby mit Weltraumsimulationen begann, hat sich zu einer professionellen Kubernetes- und Mail-Infrastruktur entwickelt. Die Projekte (paltas-api) dienen als technologisches Rückgrat für Experimente mit modernen Deployment-Methoden.
2. **Sicherheits-Pionier:** Die implementierte „Honey-Security-Architektur“ ist state-of-the-art. Durch die Kombination von NetFlow-Analyse (HoneyScan), Crowd-Intelligence (CrowdSec) und Automatisierung (AbuseIPDB) erreicht lemueIO ein Schutzniveau, das weit über dem Durchschnitt privater Hoster liegt. Das Netzwerk ist nicht nur eine Festung, sondern eine aktive Falle für Angreifer.
3. **Integrität und Identität:** Die klare Trennung von politischen Tools (13ft), Community-Interessen (Reddit) und Infrastruktur (lemueIO) zeigt eine professionelle Identitätsführung. Die Verwechslungsgefahr mit „Loomio“ ist zwar namentlich gegeben, technisch aber irrelevant.

Ausblick:
Für die Zukunft ist zu erwarten, dass lemueIO die Integration von SDR-Daten und Sicherheits-Telemetrie weiter vertiefen wird. Das Netzwerk dient zunehmend als Sensor im globalen Internet. Jeder Angriff auf lemueIO stärkt durch die CrowdSec-Integration die Verteidigung tausender anderer Systeme weltweit. Damit wandelt sich „derlemue“ vom reinen Betreiber zum aktiven Teilnehmer an der globalen Cyber-Sicherheit – ein Wächter im digitalen Rauschen.

Anhang: Detaillierte Datentabellen und Metriken (Stand Jan 2026)

Tabelle 2: Infrastruktur-Portfolio und Status

Komponente	Technologie	Funktion	Status (Jan 2026)	Sicherheitslevel
Paltas-API	Smarty, PHP, K8s	Deployment-Test, Demo	Public / Active	Hoch (CrowdSec protected)

Mail Server	Mailcow (Docker)	Kommunikation	Private / Active	Kritisch (Rspamd, DKIM)
SDR Feeder	Dump1090, Beast	Luftraumüberwachung	Active (Ports 30005/978)	Mittel (Segmentiert)
Honeypot	HoneyScan, NfSen	Intrusion Detection	Hidden / Active	N/A (Defensiv)
13ft Fork	JS, Go	Paywall Bypass	Development / Contrib	Experimentell

Tabelle 3: Analyse der Community-Interaktionen

Plattform	Rolle	Fokus-Themen	Engagement-Metrik
GitHub	Contributor / Maintainer	Mailcow, Paywalls, Kubernetes	Issues, PRs, Org-Lead
Reddit (r/derlemue)	Moderator / Creator	Philosophie, Space, Humor	~500 Member, High Retention
Reddit (Fachforen)	Expert User	DCS World, KSP, Hardware	Top 1% Badge, High Karma
Twitter/X	Kommentator	Tech-News, Ironie	Reaktiv, vernetzt

Tabelle 4: Relevante Ports und Protokolle in der Analyse

Port	Protokoll	Dienst	Relevanz für lemueIO	Risiko
30005	TCP	Beast Output	SDR Flugdaten (Export)	Datenleck bei Fehlkonfiguration

30978	TCP/UDP	UAT Raw Data	SDR (978 MHz USA/Exp)	Verwechslung mit CVE-2025-30978
443	TCP	HTTPS	Web, API, Secure Mail	Standard-Angriffsziel (DDoS)
25/587	TCP	SMTP	Mail Transfer	Spam-Relay Gefahr (hoch)

Referenzen

1. Goodbye, Harrier... : r/hoggit - Reddit, Zugriff am Januar 13, 2026, https://www.reddit.com/r/hoggit/comments/1jw1ejc/goodbye_harrier/
2. Apollo 8 - Earthrise (KSRSS) : r/KerbalSpaceProgram - Reddit, Zugriff am Januar 13, 2026, https://www.reddit.com/r/KerbalSpaceProgram/comments/1k7xmu6/apollo_8_earthrise_ksrss/
3. Issues · wasi-master/13ft - GitHub, Zugriff am Januar 13, 2026, https://github.com/wasi-master/13ft/issues
4. CVE-2021-44228 vulnerability Log4j (Solr)? · Issue #4375 · mailcow/mailcow-dockerized, Zugriff am Januar 13, 2026, https://github.com/mailcow/mailcow-dockerized/issues/4375
5. lemu-io - GitHub, Zugriff am Januar 13, 2026, https://github.com/lemu-io
6. loomio/loomio-docs: Loomio help, guides and policies - GitHub, Zugriff am Januar 13, 2026, https://github.com/loomio/loomio-docs
7. Loomio - GitHub, Zugriff am Januar 13, 2026, https://github.com/loomio
8. loomio/README.md at master - GitHub, Zugriff am Januar 13, 2026, https://github.com/loomio/loomio/blob/master/README.md
9. Loomio is a collaborative decision making tool - GitHub, Zugriff am Januar 13, 2026, https://github.com/loomio/loomio
10. 185.248.148.0/22 - bgp.tools, Zugriff am Januar 13, 2026, https://bgp.tools/prefix/185.248.148.0/22
11. docker-piaware/README.md at main · sdr-enthusiasts/docker-piaware - GitHub, Zugriff am Januar 13, 2026, https://github.com/sdr-enthusiasts/docker-piaware/blob/main/README.md
12. Predictions of Network Attacks in Collaborative Environment - IS MUNI, Zugriff am Januar 13, 2026, https://is.muni.cz/th/dmpga/thesis.pdf
13. husak/honeyscan: honeynet monitoring plugin for NfSen - GitHub, Zugriff am Januar 13, 2026, https://github.com/husak/honeyscan
14. Top Intrusion Detection and Prevention Systems for Slack in 2026 - Slashdot, Zugriff am Januar 13, 2026,

https://slashdot.org/software/intrusion-detection-and-prevention/for-slack/
15. Pricing - CrowdSec Console, Zugriff am Januar 13, 2026, https://app.crowdsec.net/pricing
16. CrowdSec Reviews 2026: Details, Pricing, & Features - G2, Zugriff am Januar 13, 2026, https://www.g2.com/products/crowdsec/reviews
17. AbuseIPDB-IP-Scanner/AbuseIPDB_ip_scan.py at main - GitHub, Zugriff am Januar 13, 2026, https://github.com/ph1nx/AbuseIPDB-IP-Scanner/blob/main/AbuseIPDB_ip_scan.py
18. Range Alerts Tutorial | User Control Panel - AbuseIPDB, Zugriff am Januar 13, 2026, https://www.abuseipdb.com/tutorial/range-alerts
19. GreyNoise Integrates with CrowdStrike Falcon Next-Gen SIEM to Unify Network Intelligence and Accelerate SOC Transformation, Zugriff am Januar 13, 2026, https://www.greynoise.io/press/greynoise-integrates-crowdstrike-falcon-next-gen-siem
20. GreyNoise Intelligence Launches Global Observation Grid to Provide Real-time Threat Intelligence on Network Attacks, Zugriff am Januar 13, 2026, https://www.greynoise.io/press/greynoise-intelligence-launches-global-observation-grid-to-provide-real-time-threat-intelligence-on-network-attacks
21. Checking It Twice: Profiling Benign Internet Scanners — 2024 Edition | GreyNoise Blog, Zugriff am Januar 13, 2026, https://www.greynoise.io/blog/checking-it-twice-profiling-benign-internet-scanners----2024-edition
22. JAX - London - Whitepaper Java Fresh Brewed 2022 - Scribd, Zugriff am Januar 13, 2026, https://www.scribd.com/document/888702340/JAX-London-Whitepaper-Java-Fresh-Brewed-2022
23. Vulnerability Summary for the Week of June 2, 2025 | CISA, Zugriff am Januar 13, 2026, https://www.cisa.gov/news-events/bulletins/sb25-160
24. The Milky Way and two Magellanic Clouds : r/Stargazing - Reddit, Zugriff am Januar 13, 2026, https://www.reddit.com/r/Stargazing/comments/1j9167x/the_milky_way_and_two_magellanic_clouds/
25. Paul is the definition of survival of the fittest. : r/pokemonanime - Reddit, Zugriff am Januar 13, 2026, https://www.reddit.com/r/pokemonanime/comments/1q2zi0g/paul_is_the_definition_of_survival_of_the_fittest/
26. Golden : r/roosterteeth - Reddit, Zugriff am Januar 13, 2026, https://www.reddit.com/r/roosterteeth/comments/1ipqbr1/golden/