Date: June 9, 2026
Subject: Security & Integrity Verification of Discord OAuth2 and Database Infrastructure
Audience: DONCOR AIRLINE Community and Passengers

OFFICIAL SECURITY STATEMENT: Integrity and Safety Architecture of the DONCOR WINGS Ecosystem

In response to recent community inquiries and proactive system routine audits, the DONCOR WINGS Management and IT Infrastructure Team has released an official technical breakdown addressing the safety, token handling, and overall architectural integrity of the system's authentication layer. This document aims to clarify for active users how passenger profile data and administration credentials are comprehensively secured against modern threat vectors.

1. Enterprise-Grade Engineering and Development

The DONCOR WINGS mileage dashboard and web environment have been designed and engineered from the ground up as a fully comprehensive, professional digital product. To guarantee an exceptional standard of infrastructure resilience, the system architecture was developed in direct collaboration with dedicated cybersecurity specialists.

Every operational pipeline—from automated data handling to high-clearance administration views—follows rigorous software design patterns. This cooperative development process ensures that the frontend application is not merely a visual dashboard, but a structurally resilient platform built according to modern enterprise security frameworks.

2. Rigorous Penetration Testing (Pentesting) Verification

Prior to its active deployment, the platform underwent comprehensive, simulated cyber-defense assessments, commonly known as **Penetration Testing (or Pentesting)**.

For our general community and frontend users, a *penetration test* can be understood as a controlled, deep-dive security inspection where cybersecurity experts deliberately act as benign, authorized attackers. These specialists utilize advanced scanning tools and exploit methodologies to aggressively test the platform's defenses. The objective is to identify and resolve potential code bugs, unauthorized access paths, or logical flaws before any real-world malicious actor can exploit them.

Through these extensive diagnostic routines, our infrastructure was continuously subjected to simulated stress tests. Vulnerabilities discovered during the initial testing phase were systematically patched and neutralized, validating the application's readiness to repel adversarial exploitation attempts.

3. The Foundation: Decentralized OAuth2 Authentication

The DONCOR WINGS mileage dashboard utilizes industry-standard Discord OAuth2 technology facilitated by the Supabase backend framework. Due to this decentralized layout, the DONCOR WINGS ecosystem **never collects, processes, or stores raw user credentials, including Discord usernames or account passwords.**

When a passenger or administrator accesses the application, authentication is securely delegated to Discord's official servers. Upon a successful verification loop, the platform receives a cryptographically signed, short-lived JSON Web Token (JWT). This process guarantees that credential theft or database leaks targeting Discord accounts via our platform are mathematically and structurally impossible.

> 🛡️ **Immutable Protection Against Password Theft**
>
> Because no authentication passwords reside within our environment, a compromise of the platform's infrastructure cannot yield actionable vectors to compromise a user's master Discord identity. The account security entirely inherits the robust multi-factor infrastructure provided natively by Discord.

4. Advanced Database Isolation: Row Level Security (RLS)

A recent deep architectural inspection confirms that the DONCOR WINGS live application employs strict database isolation barriers. The public network connection (utilizing the public client-side keys) is rigidly sandboxed from executing unauthorized write mutations.

Data Entity	Public API Rights	Mutation Vector	Architectural Protection Pattern
Flight Event Scheduling	`SELECT ONLY`	Blocked (API Level)	Protected via Row Level Security. General public users cannot forge, delay, or inject rogue flight entries. Entries are strictly limited to the verified backend pipeline.
User Profiles / Mileage Balances	`SELECT / UPDATE`	Enforced Ownership	Isolation via SQL Expression: `auth.uid() = id`. Users possess cryptographic bounds to mutate solely their personal row. Cross-profile injections are strictly denied by the server.

5. Proactive Engineering: Zero-Trust Client Rendering

To future-proof the application against cascading risks—such as "Self-XSS" (where a user attempts to break or compromise their own display name to execute rogue commands on an administrator's browser via high-clearance UI elements)—the development team is standardizing all dynamic frontend rendering layers.

By enforcing a strict migration from standard HTML data injections to programmatic text nodes, the application automatically strips and neutralizes unauthorized scripts. In combination with the server-side RLS rules, the platform establishes a comprehensive multi-tier security barrier (Defense in Depth).

6. Summary and Commitments

The DONCOR WINGS Management Team assures all passengers that their mileage accounts, virtual flight operations, and platform sessions are safely isolated according to modern corporate standards. Continuous compliance monitoring, automated vulnerability scanning, and strict role segregation will continue to be enforced to maintain the pristine reputation of the DONCOR AIRLINE simulation environment.

Issued by the authority of the Operations Hub,

The DONCOR WINGS Management Team
IT Operations & Cybersecurity Architecture Division
DONCOR AIRLINE PTFS