QuantumRand

Data Residency, Security & Entropy Pipeline Policy

Version 1.0 | Effective: March 2026 | quantumrand.dev

1. Overview

QuantumRand is a quantum entropy API that delivers true random numbers generated by IBM Quantum hardware. This document explains exactly where entropy is generated, how it travels through our infrastructure, who can access it, and what security controls are in place at each step. It is intended for security teams, compliance officers, and legal reviewers at organizations evaluating QuantumRand for use in financial, healthcare, or regulated environments.

2. Entropy Generation & Pipeline

2.1 Where Entropy Originates

All entropy served by QuantumRand is generated on IBM Quantum hardware — physical quantum processors operated by IBM at their data centers. Quantum circuits are executed on real superconducting qubits. The measurement outcomes of these qubits produce true random bits that are fundamentally unpredictable, as guaranteed by the laws of quantum mechanics. This is not simulated randomness and is not produced by a cryptographically secure pseudo-random number generator (CSPRNG).

IBM Quantum infrastructure details:

Hardware type	IBM Quantum superconducting processors
Data center operator	IBM
IBM data center region	United States (primary)
IBM compliance	SOC 2 Type II, ISO 27001
Access method	IBM Quantum REST API (authenticated)
Transport to QuantumRand	HTTPS/TLS 1.3

2.2 Entropy Travel Path

The following describes the exact path entropy takes from generation to delivery to your application:

- Step 1: Quantum circuit executed on IBM Quantum hardware in IBM data center (USA)

- Step 2: Measurement results transmitted to QuantumRand API server via HTTPS/TLS 1.3
- Step 3: Raw entropy bits received by QuantumRand backend and loaded into in-memory entropy pool
- Step 4: Entropy consumed from pool to fulfill API request (hex, integer, float, or financial primitive)
- Step 5: Formatted response delivered to customer application over HTTPS/TLS 1.3
- Step 6: Consumed entropy is destroyed — it is never stored, logged, or reused

2.3 Fallback Entropy Behavior

In the event that the IBM Quantum job queue is unavailable or backlogged, QuantumRand may blend output with a CSPRNG fallback. This condition is always disclosed in the API response via the entropy_source field:

entropy_source: "quantum"	100% IBM Quantum hardware entropy
entropy_source: "hybrid"	IBM Quantum entropy blended with CSPRNG
entropy_source: "fallback"	CSPRNG only — IBM Quantum temporarily unavailable

Customers can monitor entropy source in real time via GET /health/pool and in their audit dashboard. QuantumRand never misrepresents the source of entropy delivered.

3. Infrastructure & Data Residency

API hosting	Railway (cloud infrastructure)
Server region	United States
Database	Firestore (Google Cloud — US region)
CDN / edge	None — all requests routed directly to origin
Entropy pool storage	In-memory only — never written to disk or database
API keys	Hashed before storage — raw key never stored
TLS version	1.3 minimum on all connections
Uptime target	99.9% (current SLA — enterprise SLA available)

4. What Is and Is Not Stored

4.1 What QuantumRand Stores

- User account data: email address (hashed), account tier, created timestamp
- API key hash — the raw API key is shown once at creation and never stored
- Usage logs: endpoint called, timestamp, status code, response time, entropy source — used for billing and audit trail
- Waitlist signups: name, company, email, transaction volume estimate
- Stripe billing data: managed entirely by Stripe — QuantumRand stores only Stripe customer ID

4.2 What QuantumRand Does NOT Store

- Raw entropy bits — consumed in memory and destroyed after use
- Generated values (transaction IDs, OTPs, nonces, keypairs, signatures) — returned to customer and not retained
- Private keys from /v1/finance/keypair — shown once in response, never stored on QuantumRand servers
- Customer payload data submitted to /v1/finance/audit-sign — only the HMAC signature and payload hash are stored, not the original payload
- Payment card data — handled entirely by Stripe

5. Access Controls & Security

Authentication	API key (X-API-Key header) — required on all endpoints
API key format	qr_ prefix + 32 random hex characters
Key storage	SHA-256 hash stored — raw key never persisted
Rate limiting	Per-tier rate limits enforced at API gateway
Admin access to data	Restricted to founding team — logged
Database access	Firestore IAM roles — principle of least privilege
Infrastructure access	Railway environment variables for secrets — no plaintext config
IBM Quantum API token	Stored as encrypted environment variable — never exposed in responses

6. Audit Logging

Every API call made by a customer is logged with the following fields and is available in the customer dashboard at quantumrand.dev/dashboard:

log_id	Unique identifier for this log entry
endpoint	The API endpoint called

method	HTTP method (POST, GET)
status_code	HTTP response code
entropy_source	quantum | hybrid | fallback
response_time_ms	Latency in milliseconds
created_at	UTC timestamp — immutable

Logs are exportable as CSV via GET /v1/audit/export and are suitable for inclusion in SOC 2, PCI-DSS, and internal compliance reviews. Log entries are immutable — they cannot be edited or deleted by customers or QuantumRand staff after creation.

7. Compliance Posture

SOC 2 Type II	In progress — audit process initiated Q1 2026
PCI-DSS	Infrastructure and logging designed for PCI-DSS compatibility — formal certification in roadmap
ISO 27001	Security controls aligned to ISO 27001 framework
NIST Post-Quantum	Entropy pipeline aligned to NIST PQC standards (FIPS 203/204/205)
HIPAA	BAA available for healthcare customers on Business tier and above
GDPR	EU customer data requests handled within 30 days — contact privacy@quantumrand.dev

8. Incident Response

In the event of a security incident affecting customer data or entropy integrity, QuantumRand will:

- Notify affected customers within 72 hours of confirmed incident
- Provide a written incident report within 14 days
- Immediately rotate any compromised API keys and notify key holders
- Disclose entropy source degradation in real time via the /health/pool endpoint and dashboard

Security disclosures and incident reports should be directed to: security@quantumrand.dev

9. Contact & Questions

For questions about this document, data handling practices, or to request a Business Associate Agreement (BAA) or custom Data Processing Agreement (DPA):

security@quantumrand.dev

quantumrand.dev

This document is reviewed and updated quarterly. The current version is always available at quantumrand.dev/security.