PROJECT TITLE:

SENTINEL API GATEWAYS

R.SHRAVAN KUMAR

CYBER SECURITY

TAH3008251802 TEKS ACADEMY

TEKS ACADEMY

SHASHIDHAR

CYBER SECURITY TRAINER

Abstract

Modern API ecosystems face increasing exposure to automated attacks, unauthorized access, and malicious payloads. While real-world API gateways offer strong protection, they are often complex to configure or inaccessible for training. **Sentinel API Gateway** resolves this challenge by providing a simulated yet realistic security-driven environment where users can safely test API behaviors, attack patterns, and defensive automation.

This platform visualizes live traffic behavior, detects malicious input, enforces rate limiting, and simulates auto-ban systems. The experience mimics professional-grade gateways like Kong, Cloudflare API Shield, AWS API Gateway, and WAF behavior. Sentinel provides a hands-on, controlled cybersecurity training environment for learners and professionals.

Problem Statement

Cybersecurity learners struggle to understand how real API security systems work, especially dynamic responses like rate limiting, payload sanitization, and blocklisting. Existing tools are either **too basic** or **too advanced and proprietary**, making it difficult to perform controlled experiments.

There is a need for a **safe, interactive platform** that simulates malicious requests, traffic stress, and security responses without affecting real infrastructure.

Project Objective(s)

- Provide a simulated and interactive API security environment.
- Detect and block malicious payloads (XSS, SQLi, command injection, etc.).
- Implement configurable rate limits, risk scoring, and auto-ban logic.
- Visualize live traffic behavior through analytics dashboards.
- Allow users to simulate roles: normal user, tester, attacker, and admin.

- Provide real-time monitoring and configuration control from an admin panel.
- Serve as a cybersecurity learning platform for students and professionals.

Project Scope

- Frontend-based simulation platform (runs fully in browser).
- Supports role-based access: Client Users and Admin.
- Provides payload injection presets and custom input mode.
- Includes dashboards for monitoring traffic, logs, and user reputation.
- Optional AI-based detection using user-provided inference endpoint.
- Designed for training, learning, and demonstration purposes only.
- No real deployment onto live systems (sandbox simulation only).

🛠 Tools & Technologies

- **React 19, TypeScript, Vite** — core development framework
- **Tailwind CSS** — UI styling
- **Recharts** — live traffic graph visualization
- **Lucide React Icons** — UI iconography
- **LocalStorage** — simulated persistence
- **Optional AI Inference** — user-pluggable model endpoint

Deployment

The project is deployed and accessible online:

🔗 **Live Demo:** https://secure-api-gateway.vercel.app/

Timeline (Tentative)

Week / Day	Task
Day 1	Architecture planning, UI wireframes
Day 2	Build authentication + user roles
Day 3	Implement request simulator + traffic engine
Day 4	Add threat detection + rate limiter + auto-ban
Day 5	Create dashboard visualizations + final testing + deployment

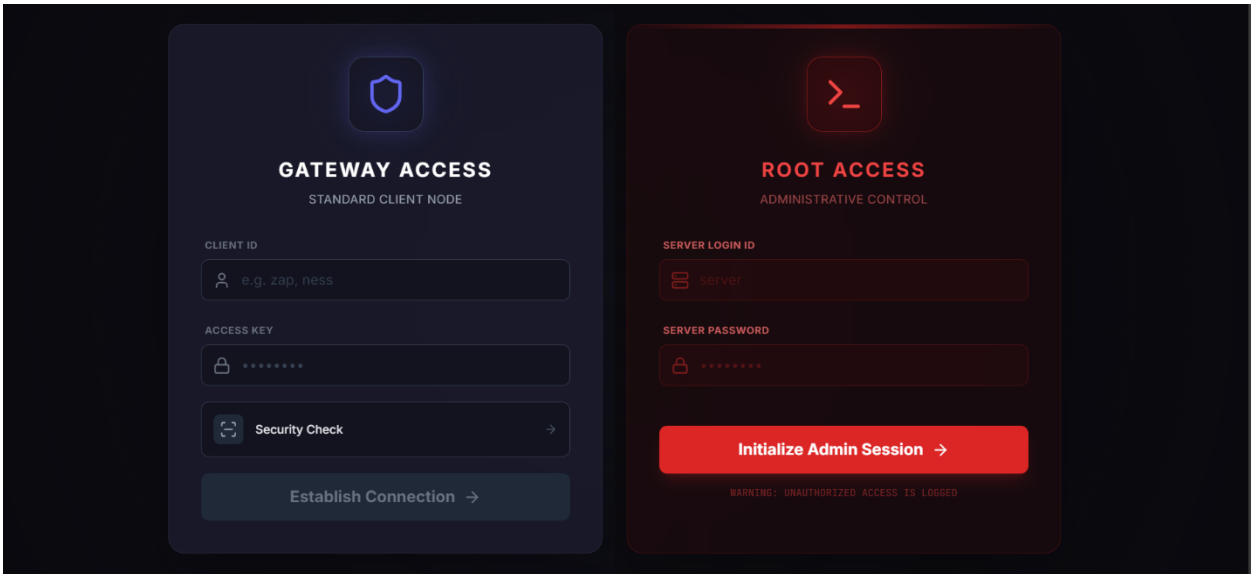
Deliverables

- Fully working cyber-security API simulator web platform
- Role-based login system (Admin + Client types)
- Live traffic visualization environment
- Threat and rate limit enforcement simulation
- Documentation and demo deployment

Project Implementation

Step 1: UI/UX Planning and Component Layout

Design modules: login, dashboard, client request panel, admin control panel, logs, analytics.



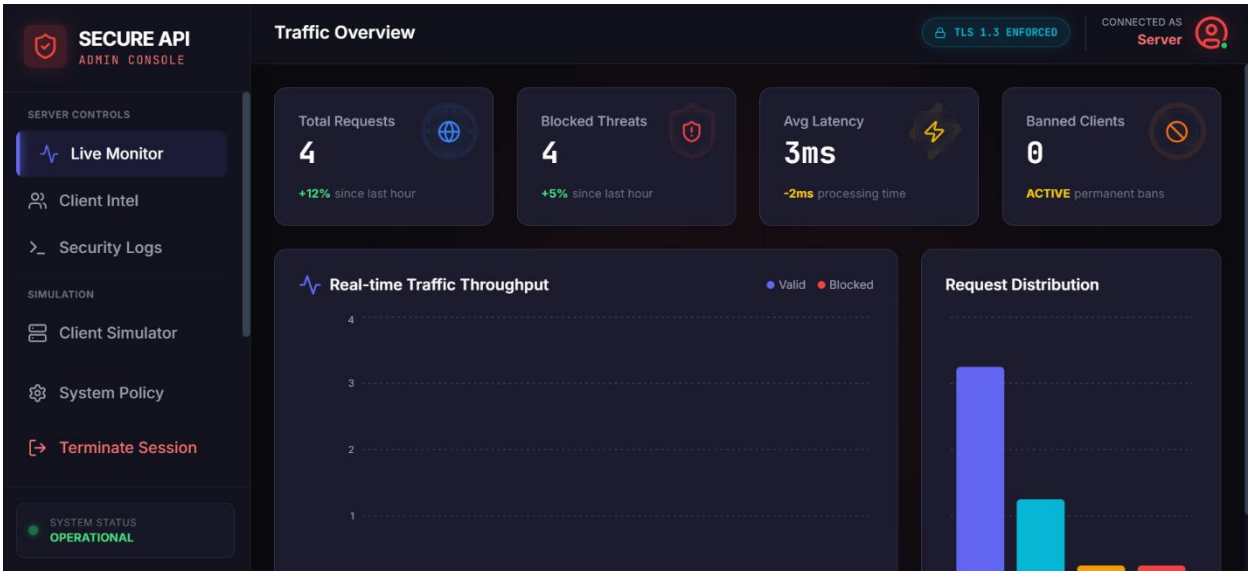
Step 2: Gateway Engine

Create simulation modules for:

- Payload analysis
- Rate limiting
- Ban logic
- Request response rules

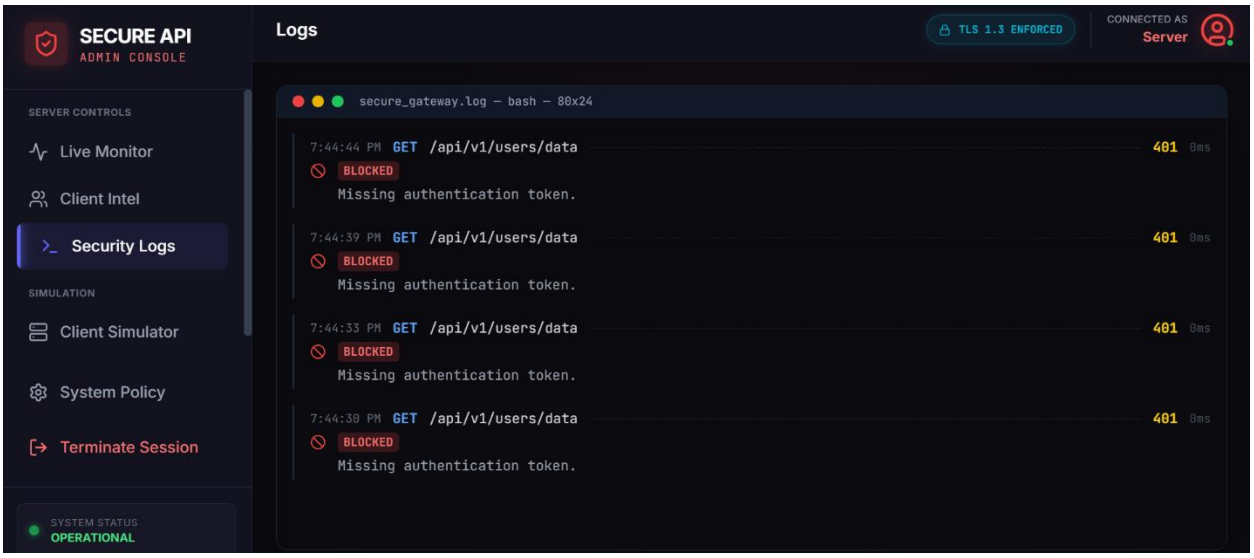
Step 3: Threat Detection Model

Implement rule-based filters and optional configurable ML inference endpoint.



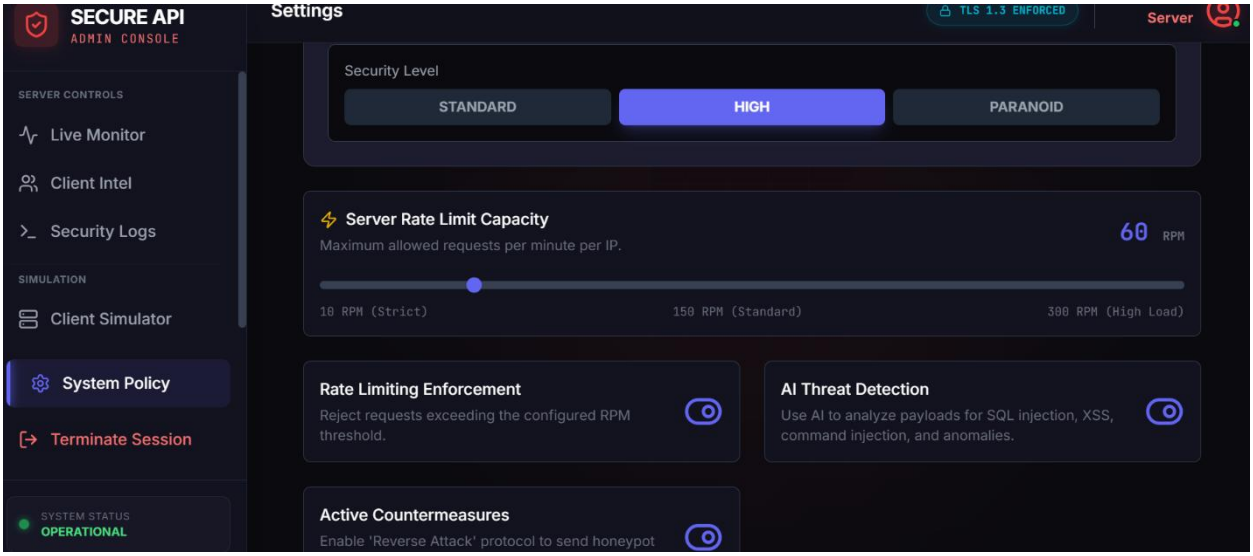
Step 4: Live Analytics Dashboard

Integrate charts and real-time updates from simulated traffic engine.



Step 5: Admin Control & System Policies

Allow configuration edits: RPM limits, security level, sanitization, and AI toggle.



User Flow

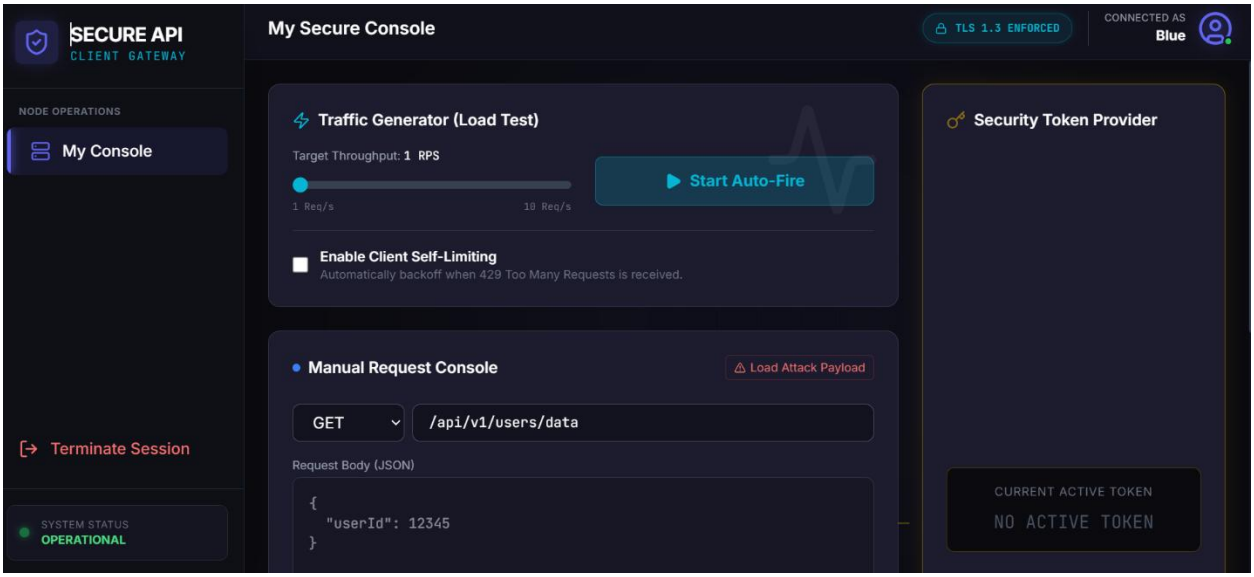
1. Login using a demo credential
2. Choose **Client Mode** or **Admin Mode**
3. Client sends normal or malicious traffic
4. Gateway responds with: 200 OK, 429 Throttled, 403 Blocked, or **Banned**
5. Admin monitors activity on dashboard and policy panel

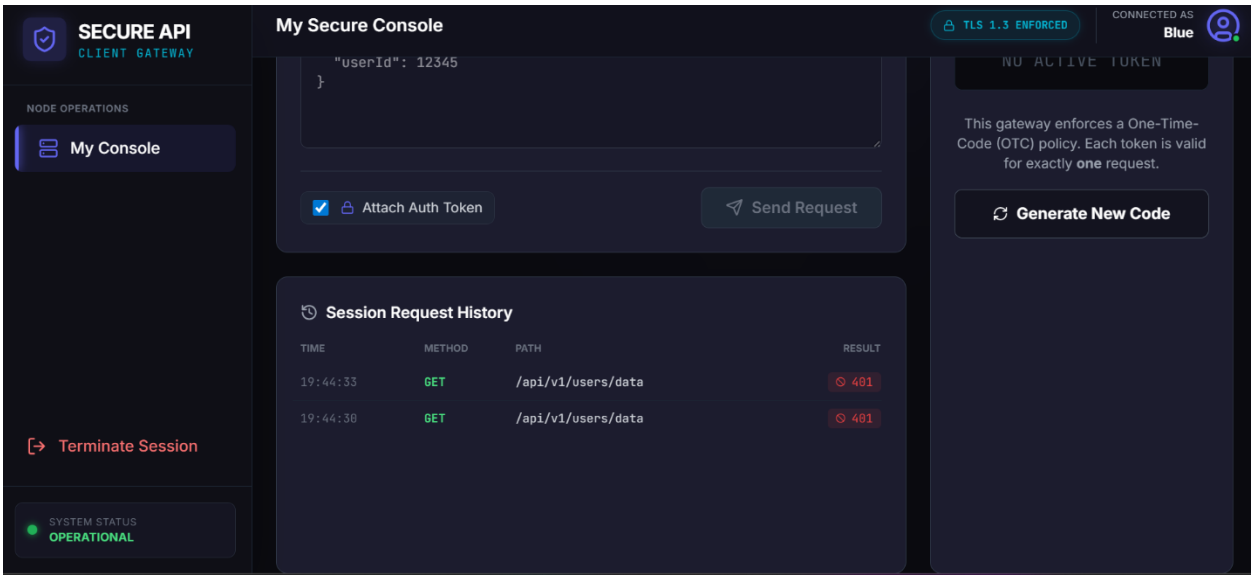
Example Use Case

Field	Value
User role	blue (red-team simulation)
Request payload	{ "username":"admin' --", "password":"1234" }

Field **Value**

System Reaction ❌ Detected SQL-Injection → Blocked → Added to risk list





Limitations & Future Work

Limitation	Planned Upgrade
No backend storage	Add SQLite/Postgres logging
Local-only persistence	Add cloud sync mode
Rule-based detection basic	Add machine learning training mode
Single user environment	Multi-node distributed simulation

Conclusion

The Sentinel API Gateway provides a powerful, safe, and interactive environment for cybersecurity education and experimentation. It demonstrates real-world API security behavior, enabling students and professionals to observe defensive automation without deploying risky requests to live systems. Its visual approach and modular design make Sentinel a valuable training and research tool.