

Leonid Sopov

Senior Software Engineer | Go, Python, Perl

Full CV: resume.sopov.org

Belgrade, Serbia

leonid@sopov.org | linkedin.com/in/leonidsopov | github.com/sopov | t.me/leonidsopov

Languages: Russian (Native), English ([Intermediate](#))

SUMMARY

Software Engineer with 20+ years in backend systems across Go, Python, and Perl.

At [MY.GAMES](#), working on secure payment infrastructure in a high-compliance environment (PCI DSS, GDPR, DORA).

At [Acronis](#), drove legacy modernization (Perl → Go), built automation tooling in Python, and contributed to backend infrastructure across multiple services.

Focused on polyglot backend systems, payment infrastructure, and practical engineering tooling.

EXPERIENCE

Senior Backend Engineer

[MY.GAMES](#) — Payment Solutions Department (Mar 2024 – Now //2 yrs)

Working on secure payment infrastructure under PCI DSS, GDPR, DORA, and internal security policy requirements.

- Maintained and refactored legacy Perl payment services for security, stability, and testability.
- Integrated new payment acquirers (Payermax, Stripe, Adyen), implementing 3DS authorization flows, IBAN/country code validation, and anti-fraud system hooks.
- Built unit, E2E, and pytest-based integration tests for payment acquirer flows.
- Maintained versioned database migrations using Alembic (Python) across multiple service databases.
- Established Docker security standards across the team: read-only containers, capability dropping, resource limits.
- Migrated services from Docker Compose to Kubernetes.
- Built and maintained the team's shared development environment: reverse proxy, uptime monitoring, mock API services, and Docker orchestration.
- Optimized stage deployment pipeline from ~9 minutes to 14 seconds: replaced git attribute filters with direct text substitution, optimized startup scripts, and added git caching.
- Implemented Git-based SSH access management with automated provisioning and account expiration.
- Fixed security vulnerabilities in payment processing code and automated secret rotation across service configuration files.

Senior Go & Python Developer

Acronis — Platform Department (Jul 2020 – Jan 2024 //4 yrs)

Worked on backend services and tooling with a focus on legacy migration, API development, and automation.

- Rewrote the Serial Number parsing service from Perl (with C++ XS bindings) into a standalone Go REST API, eliminating the C++ build dependency.
- Extended the Go Account Server: migrated API endpoints from the Python backend and resolved critical production issues.
- Migrated AMQP client library from `streadway/amqp` to `amqp091-go`, modernizing async messaging across services.
- Extended and maintained the Reporting Service: replaced DELETE-based retention with DROP PARTITION for performance, added support for new offering items, and resolved critical production incidents.
- Built a configuration parser converting multi-sheet XLSX files into structured YAML for license chain management, deployed across cloud and on-premise environments.
- Implemented E2E tests using Gherkin and Python across multiple services.

Lead of Perl Development Team

Acronis – Perl Applications Department (Nov 2019 – Jul 2021 //2 yrs)

Led the Perl backend team at Acronis (with the company since 2008), managing up to 6 developers, responsible for core internal services and infrastructure tools.

- Mentored and supported a team of two developers, ensuring professional growth and effective collaboration.
- Oversaw code quality, implemented unit testing standards, and maintained internal documentation (including POD).
- Acted as Release Manager, coordinating deployments and system updates.
- Wrote detailed design documentation and clarified undocumented system internals.
- Introduced and enforced code review practices and Perl Best Practices.
- Migrated a 122,000-line Perl codebase and containerized services using Docker.
- Served as the single point of contact for all cross-team requests and technical queries related to the Perl backend.

Founder & Solo Engineer

[Bible Online Project](#) — [only.bible](#) · [bibleonline.ru](#) · [bible.ru](#) (Oct 2003 – Now //23 yrs)

Solo-built and operated for 20+ years. Platform serving 2M+ unique users and 10M+ page views per year. Full ownership: architecture, development, deployment, monitoring, and incident response.

- Multi-layer stack: Traefik → Nginx/NJS → Apache mod_perl2 (HTML::Mason) → MariaDB + Sphinx Search + Memcached.
- Full-text search in Russian, English, and German via Sphinx Search with custom dictionaries and morphology.
- Go caching reverse proxy for users in restricted regions: hybrid SQLite + filesystem cache, content transformation pipeline, config for 35+ domains.
- Go microservice for parsing and normalizing Bible references (bible.ru): multi-language input, Unicode normalization, QR code API.
- Infrastructure: 100+ Docker containers on a dedicated server with VPS, DNS, and proxy redundancy.
- Developed a public REST API for Bible texts ([api.bibleonline.ru](#)), documented and used by external developers.
- Operates community across VK (68K+ subscribers), Facebook (7.5K+ likes), and other platforms.

SKILLS

Programming languages: Go, Python, Perl, JavaScript, SQL, bash, JSON, YAML, XML, REST API, RAML, Swagger

Technologies: Docker, Docker Compose, Kubernetes, RabbitMQ, Nginx, Traefik, DNS, Jenkins, macOS, Linux

Compliance & Regulations: PCI DSS, GDPR, DORA, Corporate InfoSec Policies

Databases: MySQL, MariaDB, SQLite, Memcached

Cyber Security: OWASP, SQL Injection, Script Injection, XXE, XSS, [CTF](#)

IDEs/Services: JetBrains, Goland, PyCharm