

Leonid Sopov

Senior Backend Engineer | Go, Python, Perl

🌐 Belgrade, Serbia

✉️ [@leonidsopov](mailto:leonid@sopov.org)

✉️ leonid@sopov.org

in [@leonidsopov](https://www.linkedin.com/in/leonidsopov)

🐙 [@sopov](https://github.com/sopov)

Russian: Native

English: [Intermediate](#)

Senior Backend Engineer, 20+ years of experience — Go, Python, Perl. Focused on legacy modernization — migrating aging systems to modern stacks while keeping production stable. Exploring AI-assisted engineering and multi-agent workflows. At [MY.GAMES](#), worked on secure payment infrastructure under PCI DSS, GDPR, and DORA compliance requirements. At [Acronis](#), held multiple engineering roles over 16 years — from Perl Developer to Head of Web Dev Team, Senior Architect, Perl Team Lead, and Go Backend Engineer. Drove legacy modernization, managed infrastructure at scale, and led backend teams.

Senior Backend Engineer / Team Lead

[MY.GAMES](#) — Payment Solutions Department (March 2024 – July 2026)

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

- Led the team in the final phase: knowledge transfer, onboarding of the successor engineers, and transition of service ownership.
- 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.
- Moved services from VMs to Docker Compose; some also went 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.
- Strengthened service security and automated configuration and secret management across payment services.

Senior Go & Python Developer

[Acronis](#) — Platform Department (July 2020 – January 2024)

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 (November 2019 – July 2021)

Led the Perl backend team at Acronis, managing a team of two developers, responsible for core internal services and infrastructure tools.

- Managed a team of two developers: code review, unit testing standards, documentation, and professional growth.
- 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 to a modern build and deployment setup, 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.

Senior Perl Developer

[Acronis](#) — Web Development Department (July 2014 – November 2019)

Maintained and developed core internal backend services. Also covered web systems administration responsibilities when the department had no dedicated sysadmin.

- Maintained and extended internal backend services: access management, licensing, and reporting systems.
- Built an access management system handling ~12,000 transactions per day.
- Replaced internal authorization with Active Directory integration, serving 1,500+ employees daily.

- Improved XML parsing performance 10x by migrating from a DOM-based parser to a custom SAX-based implementation.
- Managed infrastructure of 70+ Linux (CentOS/Fedora) servers across 2 data centers and 3 MySQL (MariaDB) clusters (acting sysadmin role); patched critical CVEs including Poodle, Shellshock, and GHOST.
- Designed and maintained deployment pipelines and release processes.

Senior Architect

[Acronis](#) — Web Development Department (August 2012 – July 2014)

Responsible for system design, reverse engineering of undocumented legacy services, and leading cross-department technical migrations.

- Designed web system architectures for internal backend services.
- Reverse-engineered legacy systems to restore undocumented business logic.
- Led cross-department migrations: software platforms, OS upgrades, Perl version transitions, and server consolidation.
- Served as primary escalation point for critical web development issues.
- Researched and recommended new technologies to improve department efficiency.

Head of Web Development Team

[Acronis](#) — Web Development Department (December 2010 – August 2012)

Led the Web Development team, responsible for backend services, team processes, and cross-team coordination.

- Managed a team of 6 engineers: weekly 1:1s, bi-annual performance reviews, team building.
- Introduced Agile methodology and deployed JIRA for project tracking (100+ new requests/month).
- Launched a unified Confluence documentation system: 7 topics, 100+ articles at launch, 30–50 new materials added monthly.
- Provided monthly development estimates for 7–10 projects.

Perl Developer

[Acronis](#) — Web Development Department (February 2008 – December 2010)

Joined Acronis as a Junior Perl Developer and progressed to Senior Perl Developer within 18 months. Developed and maintained internal backend services and automation tooling.

- Developed internal web applications, reporting tools, and administrative interfaces.
- Built automation scripts and internal tooling to support operations and content teams.
- Promoted: Junior Perl Developer (Feb 2008) → Perl Developer (Aug 2008) → Senior Perl Developer (Aug 2009) → Head of Web Dev Team (Dec 2010).

Freelance Web Developer

Campus Crusade for Christ (December 2006 – February 2008)

Development of websites using CGI/Perl.

Perl Developer

Metadesign (February 2005 – December 2006)

Web development using Perl at a web studio.

- Developed code for most of the websites managed by the studio (~20 websites over 2 years).
- Built a search engine: indexing robot, morphology-based search, and incremental page updates.
- Designed an advertising system serving over 1.5 million banners daily.
- Organized live internet broadcasting of Drag Racing events.

Founder & Solo Engineer

[Bible Online Project](#) — [only.bible](#) · [bibleonline.ru](#) · [bible.ru](#) (October 2003 – Present)

Solo-built and operated for 20+ years. Full ownership: architecture, development, deployment, monitoring, and incident response.

- Platform serving 2M+ unique users and 10M+ page views per year. Multilingual Bible reading and study platform: text reading, side-by-side translation comparison, audio, video, reading plans, Q&A.
- Perl backend (Apache mod_perl2 + HTML::Mason) handling all content delivery across multiple websites.
- Full-text search in Russian, English, and German via Sphinx Search with custom dictionaries and morphology.
- Developed a Go caching reverse proxy deployed in Russia, providing access for users in restricted regions: hybrid SQLite + filesystem cache, content transformation pipeline (URL rewriting, header manipulation, banner injection), Python-generated config system for multi-domain deployments.
- Developed a Go microservice for parsing and normalizing Bible references ([bible.ru](#)): multi-language input (Russian, English), Unicode normalization, prefix book matching, chapter/verse range optimization, QR code API with custom SVG generation.
- Developed a public REST API for Bible texts ([api.bibleonline.ru](#)), documented and used by external developers.
- Infrastructure: 50+ Docker containers across multiple servers with geo-distributed proxies; Perl and Go backends, Varnish CDN, Sphinx search, MariaDB, forum, radio streaming, REST API.
- Operates community across VK (68K+ subscribers), Facebook (7.5K+ likes), and other platforms.

Experience

- Backend Engineer – MY.GAMES (March 2024 – July 2026)

- Go & Python Developer – Acronis (July 2020 – January 2024)
- Perl Team Lead – Acronis (November 2019 – July 2021)
- Perl Developer – Acronis (July 2014 – November 2019)
- Architect – Acronis (August 2012 – July 2014)
- Head of Web Dev – Acronis (December 2010 – August 2012)
- Perl Developer – Acronis (February 2008 – December 2010)
- Freelance Web Dev (December 2006 – February 2008)
- Perl Developer – Metadesign (February 2005 – December 2006)
- Founder & Solo Engineer – Bible Online Project (October 2003 – Present)

Skills

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

Technologies: Docker, Kubernetes, RabbitMQ, AMQP, Nginx, Apache, Linux.

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

Databases: PostgreSQL, MySQL, MariaDB, Redis, SQLite.

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

Methodology: Systems Design, DevOps, Team Management, Project Management, Git, JIRA (JQL), Confluence.

AI / Agentic Workflows: Claude Code, OpenAI Codex CLI, GitHub Copilot, MCP, Multi-agent Protocols, Prompt Engineering.