Hector "lch361" Petrake

Yerevan, Armenia | author@lch361.net | codeberg.org/lch361 | www.lch361.net

Infrastructure engineer with hands-on networking experience, Linux and software engineering background and growing focus on systems engineering.

Projects

Personal infrastructure (codeberg.org/infrastructure-lch361-net) Oct 2025 - Present

Self-hosted Linux infrastructure built and designed end-to-end by me. Implemented with automation, fault tolerance, security, reproducibility, and performance constraints in mind. Since August 2026, infrastructure is released as code.

Hosted products

- Personal website (https://www.lch361.net) with comments on Fediverse.
- System performance monitoring dashboard (https://dashboard.lch361.net).
- Private & secure communication platforms (Matrix, XMPP).
- Federated social network (Fediverse).
- VPN for private network access (Tailscale).
- Automatic web content publishing for developers (Git, CI/CD).
- Authoritative DNS server for `lch361.net` zone.
- Push notifications for infrastructure events (Ntfy).
- Platform for centralized, auditable and reproducible infrastructure changes (Semaphore).

Security

- Automated daily security and bug patching across all infrastructure hosts.
- Preserved HTTPS end-to-end integrity despite potential edge infrastructure compromise.
- Protected CI/CD pipelines from executing untrusted code by verifying cryptographic signatures.

Fault tolerance

- Automated periodic data backups of all services with incremental snapshots and daily off-host replication, resulting in 1 day RPO and less than 10 minutes RTO.
- Kept all host provisioning and configuration scripts in a Git repository, enabling entire software restoration in less than 3 hours.
- Deployed private Git mirrors, allowing CI/CD to work even when upstream Git repository is down.

Performance

- Less than 2,5 minutes to synchronize web content with latest source code.
- No need for manual administration of infrastructure across 9 months.

Technologies

Hardware Raspberry Pi
OS layer Linux, ZFS.
Infrastructure Incus, OpenTofu.
Scripts & tools Ansible, Bash, shell, Python.
Software Nftables, BIND, Nginx, VictoriaMetrics, Perses, Coturn, Tuwunel, Prosody, GoToSocial, Hatsu, Jenkins, GnuPG, Headscale, Syncoid.

Jobs

System administrator Sep 2025 - Jul 2026
WEBA Ltd. *Saint-Petersburg, Russia*

Regional Internet, IPTV and IP-telephony service provider. Company website: https://weba.net

Technologies

Infrastructure Proxmox, LXC, QEMU, Docker
Servers Linux distributions - Alpine, Debian, Fedora
Switches Cisco, D-Link
Routers Juniper, Cisco, Arista, MikroTik

Achievements

- Gathered a network switch topology by developing a Python script utilizing LLDP and SNMP, allowing us to update the topology in ERP to match the infrastructure.
- Deployed and configured administrative access for network switches using RADIUS, giving our team a secure centralized login with easy onboarding/offboarding. Reconfigured hundreds of existing switches by making a Python + Telnet script which ran for 3 hours.
- Refactored a PHP API which can generate configurations for different models which have identical functionality: VLANs, LLDP, SNMP, RADIUS, security settings preventing client abuse, with configuration data pulled from ERP. Automated configuration of switches for 2 latest models.
- Investigated and resolved a complex backend issue resulting in wrong data on a production website by fixing PHP API querying billing data with SQL. Validated, tested and deployed the fix in 3 hours.
- Synchronized upstream IPTV API and billing database by developing PHP API which queries and updates both MariaDB and 3 different IPTV services in 2-3 minutes.
- Investigated the possibility of deploying IPv6 support for our clients, which led to discovery of a blocking bug in Juniper firmware's DHCPv6 implementation.
- Automated website deployment process with Git and GitLab, saving hours from the unnecessary coordination between sysadmins and web developers, leaving us both more productive.

Contributions

Alpine Linux package repository gitlab.alpinelinux.org/alpine/aports

- Contributed 2 packages to Alpine Linux, making software installation easier on my infrastructure.

Hatsu github.com/importantimport/hatsu

Self-hosted ActivityPub bridge for static sites.

- Fixed a logical bug, which was preventing Fediverse post updates for my website.

Education

State University of Aerospace Instrumentation Saint-Petersburg, Russia
Bachelor's of Science, Applied Informatics *Sep 2023 - Jul 2026*

Completed 5/8 semesters.

Technical writing

"Administration and diagnostics of Linux OS" educational manual Feb 2025 - Jun 2025

Theory, exercises and labs for an "Operating systems" course in our university.

- Organized a team of 5 students.

Policy-based routing with Tailscale www.lch361.net/articles/2026-09-03/

- Restricted Tailscale exit node access for Linux containers on my infrastructure which must not have it.

Protecting Linux servers from accidental connectivity loss www.lch361.net/articles/2026-09-06/

- Increased resilience of my Raspberry Pi SBC with no physical access to it by automatically rebooting into a minimal environment with SSH only, preventing permanent lockout caused by erroneous firewall/routing configuration.

Skills

Primary Linux, Scripts & Utilities, System Administration, Infrastructure as Code, Network Engineering, DevOps, Containerization, Virtualization, Database Management, System Monitoring, Network Monitoring, Troubleshooting

Secondary Systems Engineering, Backend Development, API Development, Database Design, Software Engineering, Technical Writing

Languages

English Fluent speech and writing
Russian Native speaker

About

The name's lch361, pronounced Lich. 🖖

I build and design IT infrastructure to solve practical problems, both in organizations and personal projects. Regardless of the task, I always aim to:

- Understand existing systems, then come up with ways to improve them.
- Improve systems incrementally and design solutions with constraints in mind.
- Understand fundamentals, then build technical solutions upon them.
- Reduce unnecessary complexity and risks both for system users and engineers.
- Solve problem first, achieve perfection after.