

# Raman Hafiyatulin (Roman Gafiyatullin)

## Contact data:

- r.gafiyatullin@me.com
- +370-693-79061
- <https://github.com/RGafiyatullin>

## About Myself

I am a software engineer with over 15 years of experience and strong background in the development of highly-available and massively concurrent network services, secure multi-party computation, such as threshold signatures and distributed key-generation protocols.

## Experience

**2022-11 — present: Working as an independent consultant**

**2023-02 — 2023-03: Copper Technologies (copper.co)** Copper is a provider of solutions for secure storage, management, and trading of digital assets.

I was working on the crypto-primitives implementation in Rust:

- distributed key generation (similar to CSI-RASh)
- threshold signature scheme for Schnorr signatures (based on FROST)
- threshold signature scheme for ECDSA (based on DKLS)

Skills: *Rust + no\_std, Public Key Cryptography, Secure Multiparty Computation*

**2022-01 — 2022-09: Parity Technologies (parity.io)**

Parity is a (leading) blockchain-infrastructure company.

I worked as an senior engineer in the team responsible for the Substrate.

Substrate — is a framework for blockchains. It provides a set of common building blocks for typical blockchain tasks: block authoring, consensus, peer-to-peer networking; while providing an extensibility mechanism through so called FRAME-framework: “user-space” code is composed into WASM-libraries.

Skills: *Rust, WASM (wasmtime), Blockchain*

**2012-10 — 2021-10: Wargaming.net**

Wargaming.net is a game development and publishing company, specialising in MMOG.

**2017 — 2021-10: Solution Architect in Wargaming Platform** Worked as part of the Architecture Working Group in Wargaming Platform.

The Platform — is a set of services common for every game (e.g. accounts, inventory, payments, campaigns etc). By using the Wargaming Platform a game development team can focus on the game itself, and reuse the existing solutions for the problems that are common to all games.

I advocated for the adoption of Rust in the development of performance critical components.

Skills: *Rust, Erlang, Scala (akka), PostgreSQL, MySQL, Kafka, RabbitMQ*

**2012-10 — 2017: Lead Software Developer in XMPP Services team** I was leading the team responsible for the development of XMPP Services for a MMO game.

The service started as a simple private-messages medium, but quickly absorbed into itself the functionality for

- rosters (friendship graph and blocklist),
- multi user chat (serving small team squads and massive game “lobby” chat),
- general purpose signalling network for the game.

The solution served the total audience of 50 million players, the typical daily peak online reached over 500K concurrently connected users (with 1.2M CCU as the record during some PR-campaign), the friendship graph consisted of over 3 billion edges.

Skills: *Erlang, Scala (akka), MySQL*

## **2006 — 2012: Early career**

Working on different projects:

- messaging solutions for cellular operators;
- billing/rating solutions for cellular operators;
- porting existing systems from .Net to Mono;
- developing desktop apps in C++, working with various multimedia formats;
- content management systems.

Skills: *Erlang, C# .Net/Mono, C++, RabbitMQ, MySQL*

## **Open Source**

### **2022 — Rabbit-Hole — A CLI for working with threshold signatures.**

Implementations of some MPC-cryptography papers, with a CLI-frontend to actually use them.

<https://github.com/RGafiyatullin/rabbit-hole>

### **2022 — Agner — actors in Rust**

My research project, in which I am looking for the possibility to use Erlang's OTP Design Principles in Rust (let it crash, supervision tree, etc).

<https://github.com/agner-rs/agner>

### **2021 — Reopenconnect — an alternative to openconnect**

An implementation of OpenConnect VPN-client (in Rust) that pretends being a Cisco AnyConnect Client.

<https://github.com/RGafiyatullin/reopenconnect>

### **2015 — Contributing into netvl/xml-rs**

xml-rs — a StAX parser for Rust. I needed this library to gracefully handle EoF when parsing incomplete documents, so I could parse XMPP-streams.

<https://github.com/netvl/xml-rs>

### **2015 — Contributing into uutils/coreutils**

uutils/coreutils — is an implementation of coreutils in Rust. I implemented unix `expr` utility there :)

<https://github.com/uutils/coreutils>

### **2015 — Orca — A better MySQL client for Erlang**

<https://github.com/RGafiyatullin/orca>