

Evard Vadim

v.e.evard@gmail.com

+7-921-565-14-32

<https://evard.online>

<https://github.com/Pipeliner>

skype: evardvadim

Birth: 13.01.1992

Experience

2019-2020: [Hopper](#), software developer. Python, automation of physical and virtualized Android devices, Ansible, networking. Improved overall service stability (from about 30% of failed tasks to less than 1%) and introduced new required features.

2018-2019: NDA. Blockchain infrastructure development. Virtual machine and toolchain.

2017 - 2018: freelance work. Blockchain development: Ethereum (Solidity, web3.js, Truffle), Bitcoin (proprietary database app maintenance, C++), Apla. Telegram bot development (Python, [@anti228bot](#), [@m_divisionbot](#)).

2016 - 2017: First Line Software, C++ developer. Porting system for instant photo delivery from photographer to publisher to an embedded Linux device.

2015 - 2016: DrWeb, C++ developer. Firewall for Windows. C, C++, Windows service and a kernel WFP driver. User and kernel mode debugging with WinDbg, including automation with PyKd/Python. Dump analysis. Reverse Engineering of network applications. Visual Studio, Git, Gitlab, Mantis.

2015 - 2017: managing LessWrong community meetups in Saint Petersburg (<https://www.meetup.com/St-Petersburg-LessWrong-Meetup>). Talks, games, exercises, managing speakers.

2010 - 2014: successful participation in CTF information security contests in SPSU team PeterPEN [\[1\]](#), [\[2\]](#).

SPbSU, summer school 2011 - study on RFID technologies and associated dangers. Demonstration of typical buffer overflow vulnerability and its successful exploitation. Report on various vulnerabilities possible with RFID.

Education

2018: MITx 16.00x, Introduction to Aerospace Engineering: Astronautics and Human Spaceflight.

2009 - 2014: St. Petersburg State University, Faculty of Mathematics and Mechanics,
Department of System Programming.

Skills

Advanced Python, C and C++ development (including Qt GUIs, for Linux and Windows).

Knowledge of typical software vulnerabilities (OWASP Top 10), mitigation and basic exploitation techniques.

Ethereum contracts and simple web/CLI UIs (Solidity, JavaScript).

Advanced Linux user (including building software from source code and service administration).

Docker and Docker Compose: some basic usage (simple Dockerfiles, running services etc).

Jenkins: configuring build jobs for several C++ projects.

Ansible: basics of provisioning and configuring services.

Basic knowledge of machine learning and neural nets (Tensorflow, sklearn).

Languages

Russian - native.

English - advanced/C1.

Spanish - basic.