Samira C. Oliva Madrigal

Personal Site
GitHub
LeetCode Submissions
LinkedIn
Email: scolivamadrigal@gmail.com
Cell: +1415.794.3691

CAREER INTERESTS

- Cryptography (PQC), Networking L5/3 Security & IP Routing Algorithms & Protocols, Quantum Computing

RELEVANT COURSEWORK

- TTL Logic Gate Design, Digital Design (Verilog), Computer Architecture and Design (MIPs), Advanced Computer Design (Verilog), Application-Specific Design for Cryptosystems (Verilog/SystemVerilog), Information Security, Embedded-System Design, Microprocessor Design, Real-Time Embedded System Co-Design, Algorithms and Data Structure Design (C/C++), Advanced Algorithm Design (C), System Software (C), Operating System Design, Compiler Design (x86), Software Engineering, Software Quality Assurance and Testing, Software Security Technologies, Computer Networks, Computer Network Design, Network Security, Network Architecture and Protocols, Network Programming and Applications, Advanced C Programming, C++ for C Programmers, Server-Side Web Programming, x86 Assembly Language, UNIX/Linux, Shell Scripting, Numerical Analysis and Scientific Computing, Linear Algebra, Calculus-based Physics (Mechanics, E&M, Optics & Waves, & Particle)

TECHNICAL SKILLS

- **Areas**: **Applied Cryptography, Internet TCP/IP suite, Post-Quantum Cryptography**
- **Work**: System Design, Prototyping, Validation, & Testing in software and hardware
- **Domains**: Spans expertise in hardware, software, & firmware domains.
- **Research:** Index-calculus, Cryptography, Post-Quantum Cryptography, Blockchain
- **Applied Math & Physics**: Field arithmetic, proofs, problems and instances of problems on which crypto constructions are built, linear algebra, statistics, probability distributions, FFTs, calculus, differential quations, interference, parallelism, quantum computing
- **Cryptography & Protocols/Algorithms**: Modern, quantum, and post-quantum cryptographic primitives and schemes, sieving, cryptanalysis, block cipher constructions and analysis, cryptographic hash functions, MACs, digital signatures, HMACs, algebraic constructions, rings, modular multipliers, hash-based, lattice-based, code-based, multivariate-based, supersingular-ec, rank-based, PKC, symmetric, secret sharing, KEX, PKI (RFC4949), OWFs, fields, IFP, DLP, ECDLP, NP problems, SIS, SIVP, HPP, SVP, LWE, R-LWE, RSD, oil + vinegar, nonlinear multivariate systems of equations, NP-hard, Montgomery, Blakely, BMM, interleaved multipliers, Fiat-Shamir, DES, AES, RSA, ECC, DH, KECCAK, *x.509*, ECDH, SPHINCs, Rainbow, McEliece, QC-McEliece, NTRU, CFS, Isogeny-based ECC, SIDH, KECCAK, MACs, HMACs, quantum random number constructions, models
- **Information Security**: confidentiality, authentication, integrity, secure coding, scanners, viruses, hardware bugs, side-channel analysis, speculative execution, constant-time algorithms, gadgets, ROP/JOP, control-flow attacks, remote code execution, DDoS, oracles, buffer overflows, code injections, sniffers, backdoors, cloud, hypervisors, deep web
- **Networking & Protocols/Algorithms**: Signal processing, QAM-64, symbol/bit encoding schemes, error-correction, MACs, CD-MAC, CA-MAC, ARP, NDP, STP, IEEE 802.3, IEEE 802.11x, PPP, Tunneling, VNPs, VLANs, QoS, IP (v4/v6), CIDR, RFC 1918, MPLS, Multicast, PIM (sparse, dense), IGMP (v4), MLP (v6), IPSec, NAPT, ICMP/v6, DNS, TLS, TCP, UDP, DIJKSTRA, OSPF, IS-IS, iBGP, eBGP, inter-AS routing, intra-AS routing, switching fabric, SDNs, control plane, data plane, Cloud (I/S/P/B as a Service), containers, microservices, sockets, Network OS (e.g., IOS XR), packet analysis
- **Digital & Analog Design**: Combinational & Sequential Circuits; System Memory; Embedded System Design, FreeRTOS, Raspbian, microncontrollers with ARM cortex, communication protocols (UART, CAN, I2C, etc.), device drivers; assembly language, different microprocessors, LCPExpresso
-**Implementations**: cryptographic algorithms (sw & hw), FSM, pipelining, x86 compiler, processor, hardware verification with test vectors, software development, automated testing of Internet protocols
- **Programming**: **C pointer-based language**, systems programming, OOP, C++, Java, HDLs: **Verilog/SystemVerilog**, ISAs: RISC (MIPs) and CISC (x86), Scripting: **Python**/Shell Scripting (bash, tcsh, bourne shell), Multithreading, Concurrency, Parallel Processing (with Python Ray), Virtualization

- **Computer Science**: linear, non-linear, & dynamic data structures (e.g., trees, forests, and graphs), red-black, merkle trees, m-way trees, dynamic programming, complexity theory, space and time algorithmic complexity analysis, hardware analysis (CC count, cell count, critical path delay)
- **Industry tools**: **Vivado**/ISE, **FGPAs** (Nexsys3, COM-1800, Virtex7), **Xcode/PyCharm**/Eclipse/gcc, Visual Studio/MIPs Assembler, MATLAB, Pytest, TextFSM, Wireshark for packet analysis, routers (ASR9K, NCSxx), switches, line cards, Spirent/Ixia traffic generators, testbed setup, Jenkins, VMs, OS: MacOS, UNIX/Linux/Windows, Fedora, Ubuntu
- **Public/learning tools**: Cisco Dcloud, Amazon VPC, GNS3, IBM Quantum/Qiskit, virtual classrooms
- **Familiar with**: MPKC, SNARKs, MPC, ZK proofs, succinct arguments, Go, DAPPs in Solidity, ARM TrustZone, Docker & Kubernetes, building a container from scratch, FIPS-140-3 and related ISO standards, HSMs, PIN cracking, Payment Card Industry (PCI) Security Standards (e.g., Crypto Key Blocks)

KEY FACETS

- Self-starter, likes to benchmark work against state-of-the-art, fast learner, works excellent in group or individual

EDUCATION

2021 **University of Buenos Aires (virtual ECI34), Argentina**
Certificate of Achievement - Quantum Random Number Generators.

2019 - 2018 **San José State University, San José, CA**
M.Sc. Computer Engineering with 3.571 GPA
Double Specialization: Networking Systems & Secure Systems
Thesis: *Reduction-free Multiplication in GF(2^n) Applicable to Modern and PQC schemes*
◆ Fully-interleaved Montgomery-type product.
◆ Tested with FIPS 186-4 ECDSA curves, medium and large fields, and Mersenne curves.
◆ Bit-parallel hardware simulations in Verilog exploit true parallelism and computes in exactly half the clock cycles ($n/2$) for an n-bit binary field; also purely combinational logic with CC set by FSM, and higher-radix implementations.
◆ Results and trade-offs are similar to BMM but without the reduction circuits.
◆ The software simulation in Python was done using task-level parallelism.
◆ Incorporation with other schemes and radices may lead to significant speed up of existing and new cryptographic schemes.
◆ Applications to PQC schemes, particularly lattice-based schemes.

2013 - 2017 **San José State University, San José, CA**
B.Sc. Computer Engineering, Minor Computer Science with 3.362 GPA
Senior Project: *FPGA-based Blockchain Accelerator for Ethereum Proof-of-Work*
◆ Contributed to hardware design and implementation of Keccak based on NIST FIPS 202
◆ Contributed to all research, testing, and results analysis for Ethereum, blockchain, and the Xilinx Memory Interface IP

2010 - 2013 **San José State University, San José, CA**
A.A. Systems Programming with 3.46 GPA; French & Italian Studies with 4.0 GPA
◆ Engineering, Math, Science, UNIX/Linux, Shell Scripting, Operating System Internals, Advanced C programming, C++, Java, Web Development, & x86 assembly.

RESEARCH EXPERIENCE

2019 **San José State University, San José, CA**
NSF Post-Quantum Cryptography Proposal

2019 **San José State University, San José, CA**
Modular Multiplication in $GF(2^n)$

RELEVANT PROFESSIONAL EXPERIENCE

Fall 2019 **San José State University, San José, CA**
Instructional Student Assistant
- Assisted professor for graduate course in Cryptography and Network Security.
- Prepared review notes for students. Worked grading homework assignments, quizzes, & exams.
- Galois Field Arithmetic, Public-key & Symmetric-key Cryptosystems, Digital Signatures, Authentication, Kerberos, PKIs, Certificates, L5/3 Security Protocols

2018 - 2017 **Cisco Systems, Inc., Milpitas, CA**
Software Engineer
- Feature testing and automation for next-generation Service Provider.
- Tested protocols of network operating system on different router platforms.
- Code Review, bug resolution with DEs, regression testing, and mentored a remote colleague.

ACADEMIC PROJECTS

2019 Steganography Python Application with TLS (OpenSSL, virtual datastore, & sockets programming)
2019 Public-Key Infrastructure Application using x.509 certificates
2019 Index-Calculus Research Project
2019 Topology testbed setup and in-depth analysis of L2-L4 Internet protocols
2018 Shodan Port Scanning Research Project
2018 Network Enterprise Project on Embedded Devices
2018 - 2017 Testing & Automation for CPU Infrastructure, BSP, & IEEE 802.3ad
2017 Numerical Methods to Approximate IVPs
2016 Hardware Implementation of AES based on FIPS-197
2015 32-bit Pipelined MIPs Processor
2014 Crypto Workhorse: Block-Cipher Study with Focus on AES and DES

AWARDS & HONORS

2019 Best Homework for graduate course in network programming and applications.
2017 Cisco You Inspire 2 Award - Energetic engineer who takes up lab activities.
2017 Dean's Scholar - 55th annual Honor's Convocation for GPA of 3.64+ for 2+ contiguous semesters.

TEACHING

I am interested in teaching undergraduate and graduate courses in: Computer Networking, Programming in C (Introductory, Data Structures Design), Advanced Algorithm Design, Cryptography and Network Security, Information Security, Digital Design in Verilog, Precalculus, Integral & Differential Calculus

LANGUAGES

- Excellent written and verbal communication skills.
- Native: English, Spanish; Full professional working: Italian; Professional working: French; Beginner: Russian.

ACTIVITIES

- IACR, EITCI, Volunteering at St. Lucy Catholic Parish, Cryto and Internet Research & Development, Reading, Gardening, Running, Natural Languages, Mentoring (English, STEM, & career advising)

AVAILABILITY

- I am always open to exploring promising work and collaboration opportunities.
- 2020 and part of 2021 were interrupted but remained active with research and development in crypto and networking areas, implementing in Python/C and Verilog. Learned the fundamentals of quantum computing, Quantum Protocols and Algorithms, solved chapter exercises, and ran simulations of all said algorithms on different backend simulators and real quantum hardware through IBM's Qiskit textbook and Python simulators (jupyter-lab and jupyter-notebook).