Teaching the Next Generation of Cryptographic Hardware Design to the Next Generation of Engineers

Aydin Aysu
Department of Electrical and Computer Engineering
North Carolina State University, Raleigh, NC, U.S.A
aaysu@ncsu.edu

ABSTRACT

Evolving threats against cryptographic systems and the increasing diversity of computing platforms enforce teaching cryptographic engineering to a wider audience. This paper describes the development of a new graduate course on hardware security taught at North Carolina State University during Fall 2018. The course targets an audience with no background on cryptography or hardware vulnerabilities. The course focuses especially on *post-quantum cryptosystems*—the next-generation cryptosystems mitigating quantum computer attacks—and evolves into designing specialized hardware accelerators for post-quantum cryptography, executing sophisticated implementation attacks (e.g., side-channel and fault attacks), and building countermeasures on such hardware designs. We discuss the curriculum design, hands-on assignment's development, final research project outcome, and the results obtained from the course together with the associated challenges. Our experience shows that such a course is feasible, can achieve its goals, and liked by the students, but there is room for improvement.

KEYWORDS

education, hardware security, post-quantum cryptography, FPGA

ACM Reference Format:

Aydin Aysu. 2019. Teaching the Next Generation of Cryptographic Hardware Design to the Next Generation of Engineers. In *Great Lakes Symposium on VLSI 2019 (GLSVLSI '19), May 9–11, 2019, Tysons Corner, VA, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3299874.3317994

1 INTRODUCTION

The supply for cybersecurity specialists does not scale with the demand, causing what is known as the *cybersecurity talent gap* [12]. Indeed, the unfilled cybersecurity job openings will triple by 2021 reaching to 3.5 million [6]. There are two main causes for this gap. First, capabilities of adversaries are disproportionately increasing—cybersecurity failures are expected to double in five years causing $6 trillion annually [6]. Each new vulnerability adds a layer of complexity to the defense framework, which has to mitigate all attacks in the threat model. Second, especially with the rise of the Internet-of-Things (IoT), more computing devices surround us and they are all connected, widening the surface of exploits.

Among cybersecurity exploits, hardware attacks are of growing concern because trusted computing in hardware is fundamental for all information security practices. The basis of security guarantees in digital systems boils down to a set of cryptographic operations executing in a hardware root-of-trust. When this root is compromised, security enforcement mechanisms at higher abstraction levels will inevitably fail. Driven by political or economic agendas, motivated entities are thus targeting the hardware layer with ever-advancing attacks [16]. Meanwhile, such systems are deployed in mobile and IoT context that have limited resources to enforce security.

To take on the daunting task of securing cyberinfrastructure under such circumstances, we need to train a large number of hardware security specialists. This paper describes the development of a new graduate course on hardware security with an emphasis on next-generation cryptosystems. The objective of the course is to provide a breadth of understanding of hardware security and an in-depth comprehension of the implementations of cryptographic hardware, potential exploits, and associated defenses. The challenge of such a course is to cover related concepts in hardware design, applied cryptography, computer architectures, and statistics, among others, while balancing the breadth and depth of relevant subjects.

The method we used was to first introduce the basics of applied cryptography and interleave the teaching of implementation attacks in between as they fit. We design a set of assignments for students to analyze implementation attacks and apply them on real cryptosystems through hands-on experiments. This is accompanied with presenting and discussing key papers on related topics. The course gradually evolves into a self-proposed final research project that tackles an open-ended problem. The breadth is covered towards the second half of the course through instructor-led lectures.

We argue that every electrical and computer engineering department should include a hardware security course. Although such courses have been taught in some universities [8, 9, 11, 13, 15, 17, 20] in US, none have (based on public information) our focus on next-generation cryptographic systems. We share our experience in developing this course and provide useful feedback for future adopters.

2 PRELIMINARIES

The title and hence the subject of our course is Cryptographic Engineering and Hardware Security. Hardware security is a broader concept that goes beyond crypto-engineering and studies various ways the hardware can lose trust. While we do recognize such issues, e.g., in hardware Trojans, phyiscal unclonable functions, logic locking, etc., and teach them to provide a breadth of knowledge, the course has a certain focus on efficient and secure implementation of cryptography and specifically post-quantum cryptography. This section provides a background on related subjects.

2.1 Implementation Attacks

The standard model of cryptography assumes that the adversaries are limited by the amount of time and computational power available. Under this model, ciphers are build with the premise of making theoretical cryptanalysis computationally infeasible. The standard model, however, does not capture implementation attacks. Such attacks can indeed extract secret information by exploiting hardware behaviors of the computing device while it processes cryptographic functions. These attacks are not necessarily the result of *bad* implementations per se, but they naturally occur due to the way hardware

GLSVLSI '19, May 9–11, 2019, Tysons Corner, VA, USA

ACM ISBN 978-1-4503-6252-8/19/05…$15.00
https://doi.org/10.1145/3299874.3317994

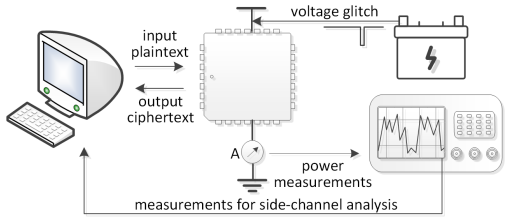

Figure 1: Block Diagram of Implementation Attack Setup

fundamentally operates. Straightforward realizations will thus be vulnerable to such attacks unless there is a specific defense.

Implementation attacks are broadly categorized into invasive and non-invasive attacks. While invasive attacks change execution behaviors to achieve its goal, non-invasive attacks simply rely on observing/analyzing execution-related side-effects.

2.1.1 Non-Invasive Attacks: Side-Channel Analysis. Side-channel analysis extracts secret information from a physical platform by observing unintentional implementation effects of computations. These attacks break cryptosystems without breaking the mathematical structure of the encryption but instead by analyzing leaks that are correlated with secret key values. Power consumption is major source of hardware vulnerability because CMOS power is data dependent—when a secret key is processed, there is an inherent dependency of the key values and the resulting switching activity.

Simple Power Analysis (SPA) is commonly referred to the types of attacks that abuse control-flow variations generated from key-dependent branches. These attacks observe large differences in power characteristics that distinguish a certain sequence of operations from another [10]. *Differential Power Analysis* (DPA), by contrast, can extract small data-flow variations by analyzing many power measurements obtained under the same secret key [10].

2.1.2 Invasive Attacks: Fault Attacks. Fault attacks perturb the device behavior and deduce the secret key from faulty executions. The classic example is the attack on RSA signatures where a single fault inserted during modular exponentiation can cause a faulty output that leaks private keys [3]. There are various ways to inject faults into computing devices and exploiting faulty behavior [21].

2.1.3 Experimental Setup. Figure 1 outlines a typical experimental setup for side-channel and fault attacks. In side-channel scenario, the target device storing the secret key executes cryptographic operations and the oscilloscope captures the power consumption during these computations. Measurements are then returned with the associated input/output to apply the side-channel attack. In fault attack scenario, the supply voltage is glitched for a short period causing a faulty output, which is analyzed to extract the secret key. Figure 2 depicts the board used in the course to apply side-channel and fault attacks. This setup uses the SAKURA-G evaluation board [7], which includes (circled in white) a Xilinx Spartan-6 FPGA for cryptographic processing, an SMA connection to measure FPGA power consumption, and an external voltage supply pin for voltage glitching.

2.2 Post-Quantum Cryptography

Existing public-key cryptosystems like Diffie-Hellman, RSA, DSA, and ECDSA, rely on the difficulty of solving integer factorization or discrete logarithm problems, which are *not* conjectured to be NP-hard and which have polynomial-time quantum solutions [14, 18]. Post-quantum cryptography seeks alternative cryptosystems that are secure against quantum cryptanalysis. These are still classical algorithms that operate on classical computers but they rely on

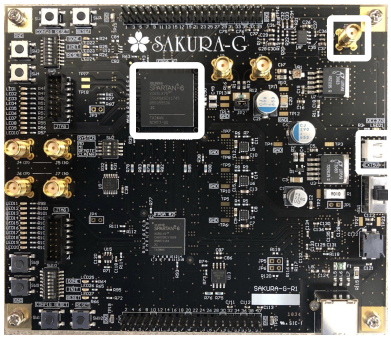

Figure 2: The SAKURA-G Hardware Security Platform

different problems. Among such proposals, *lattice-based cryptography* is a major candidate in which the security can be reduced to NP-hard lattice problems such as the shortest and closest vector problem [1, 5]. These problems, like other NP-hard problems, cannot be exponentially accelerated by quantum or classical computers.

3 COURSE STRUCTURE AND OVERVIEW

This section gives an overview of the course and describes its structure and schedule. The course targets students with hardware/software coding background—since graduate students typically know how to design software, setting an HDL course as a prerequisite is sufficient. The course does not expect prior knowledge of cryptography or related hardware security issues.

3.1 Course Objectives

The primary objective of the course is to develop an understanding of the various issues related to cryptographic engineering and hardware security. We set the learning objectives as follows.

- Develop cryptographic implementations from a given algorithmic/mathematical description and apply prevalent optimization techniques for embedded deployment
- Demonstrate major implementation attacks on cryptographic hardware, and design efficient and effective countermeasures
- Identify state-of-the-art cryptographic primitives and evaluate their use for novel applications
- Analyze key components of trusted computing platforms and vulnerabilities of modern computer architectures

3.2 Course Structure

We structure the course with 5 components. We intend to cover both the depth and the breadth of topics through different components—while **regular lectures** provide the breadth, the rest explore the depth. This structure is very similar to a course taught at Virginia Tech [17]. Our main difference is the focus on post-quantum cryptography. The five course components are summarized as follows.

(1) Regular Lectures are taught by the instructor or a guest lecturer from academia or industry and cover various topics on hardware security (details in 1). **(2) Course Assignments** are hands-on experiments to design cryptographic hardware and to demonstrate implementation attacks/defenses. **(3) Paper Presentations** allow students to study and present important related concepts not covered during lectures—selection of papers is completed early in the course. **(4) Paper & Presentation Reviews** motivate students to read the papers prior to their presentation; several students are randomly selected at the beginning of the course to review the presenter and the paper. **(5) Final Projects** are self-defined, open-ended research projects testing the usability of student's developed expertise for the future. At the end of the course, students present their final projects and prepare a report in a publication format.

Table 1: Course Schedule for Fall 2018

Week	Days	Topics	Deadline
1	23 Aug	Introduction & Course Overview	
2	28/30 Aug	Symmetric-Key Cryptosystems	
3	4/6 Sep	Side-Channel Analysis (SCA) – Part I	Paper Select
4	11/13 Sep	No lecture: Inclement Weather	
5	18 Sep	Public-Key Cryptosystems	Assignment
5	20 Sep	Side-Channel Analysis – Part II	I
6	25 Sep	Hash Functions	
6	27 Sep	Paper Presentations (SCA Attacks)	
7	2 Oct	Post-Quantum Encryption – Part I	
7	4 Oct	No lecture – Fall Break	
8	9 Oct	Paper Presentations (SCA Defenses)	Assignment
8	11 Oct	Random Number Generators (RNGs)	II
9	16 Oct	Paper Presentations (RNGs + Bit-Slice)	
9	18 Oct	Post-Quantum Encryption – Part II	
10	23 Oct	Physical Unclonable Functions (PUFs)	
10	25 Oct	Paper Presentations (PUFs)	
11	30 Oct	Fault Attacks and Countermeasures	Assignment
11	1 Nov	Lightweight Cryptography	III-A
12	6 Nov	Paper Presentations (Faults + Lightweight)	Assignment
12	8 Nov	Hardware Trojans (HT) and Backdoors	III-B
13	13 Nov	Project Phase I Presentations	Project
13	15 Nov	Trusted Computing Bases (TCBs)	Phase-I
14	20/22 Nov	No lecture – Thanksgiving	
15	27 Nov	Paper Presentations (HT + TCBs)	
15	29 Nov	Micro-Architectural Attacks/Defenses	
16	4 Dec	No lecture – Instructor Unavailable	
16	6 Dec	Paper Presentations (Micro-Arch. Attacks)	
17	18 Dec	Finals: Project Phase II Presentations	Phase-II

3.3 Course Schedule

Table 1 shows the course schedule and the timing of course activities. The course is designed for the 17-week academic semester of the university. The final's week are allocated for the presentation of the final project's results. Paper presentations, which cover an important concept related to the lectures, take place as close as possible to the associated lecture. One challenge with this tight schedule is to leave some redundant time in between to cover instructor unavailability and inclement weather.

3.4 Student Background

In the first lecture, we conducted a short survey to assess the background and interests of course participants. The survey consisted of 5 written questions, some asking for a brief description of their background on hardware and cryptography. The class returned 29 filled responses and the majority were anonymous. The mean scores for hardware and cryptography background was respectively 3.1 and 1.4, which meet our expectations. All but 6 students reported that they took at least one course on hardware design; 8 reported that they have industry experience. By contrast, only 5 students have reported a prior knowledge of cryptography. To further probe student's hardware background, we conducted another survey in the second lecture that consisted of 5 basic questions on digital design and HDL. The results align with the self-evaluations: out of the 30 completed reports, 5 students showed a lack of fundamental understanding. Unfortunately, the anonymity of the survey responses does not allow correlating course outcome to prior background.

4 ASSIGNMENT DEVELOPMENT & RESULTS

Cybersecurity education should not be purely theoretical. Assignments play a major role in this course as they test the understanding of the core concepts and their real-world applicability. All assignments are hands-on experiments that require building and/or breaking cryptosystems. Assignments prepare students for the final project, where they will use such skills on an open-ended problem.

4.1 Assignment I: Differential Power Analysis

The purpose of the first assignment is to provide a practical introduction to power-based side-channel analysis. The assignment asks to apply DPA on the Advanced Encryption Standard (AES). The students are given 10000 power measurements (i.e. time plots, traces) taken during AES Electronic Codebook (ECB) executions running on SAKURA-G with the associated input/output values of AES, and are asked to extract the 128-bit unknown secret key via DPA. Note that the theoretical security of this AES is 2^{128} yet the student's goal is to break it in 10000 tests. The students are also not given the AES hardware design but instead require to reverse-engineer the design parallelism from the power profile. Prior to this assignment, DPA and AES are covered in class. The assignment walks students step-by-step through the attack procedure.

We believe that this assignment is a good introduction to cryptographic engineering because it shows how a real implementation using the current encryption standard can truly be vulnerable to implementation attacks in practice. Figure 3 shows the attack results from a submission. All 128-bits (16 bytes) of the secret key are indeed vulnerable to DPA—in all cases, the correlation trace for the correct key byte guess reveals a distinguishable relation with power compared to incorrect guesses. The submissions show that 24 out of 28 students were able to successfully conduct the attack.

4.2 Assignment II: Hardware Design of Post-Quantum Public-Key Encryption

The purpose of the second assignment is to let students study a cryptographic algorithm in detail and develop its hardware implementation with proper optimizations. The assignment requires designing an FPGA implementation of the lattice-based Ring-Learning-with-Errors (R-LWE) post-quantum public-key scheme with binary errors. Prior to the assignment, public-key encryption and lattice cryptography (but not the specific algorithm) is covered in the lectures. The students, however, are given very little guidance this time. The assignment simply points to the reference paper [4] and provides a testbench for hardware output verification. The testbench I/O is strictly enforced in the assignment. To impose design-space exploration, half of the groups are forced to design for maximum throughput and the other half to minimum area.

Figure 4 illustrates the outcome of Assignment II. 8 out of 11 groups (of two students) were able to design a stable and functionally correct hardware. The figure summarizes the area-cost (in LUT

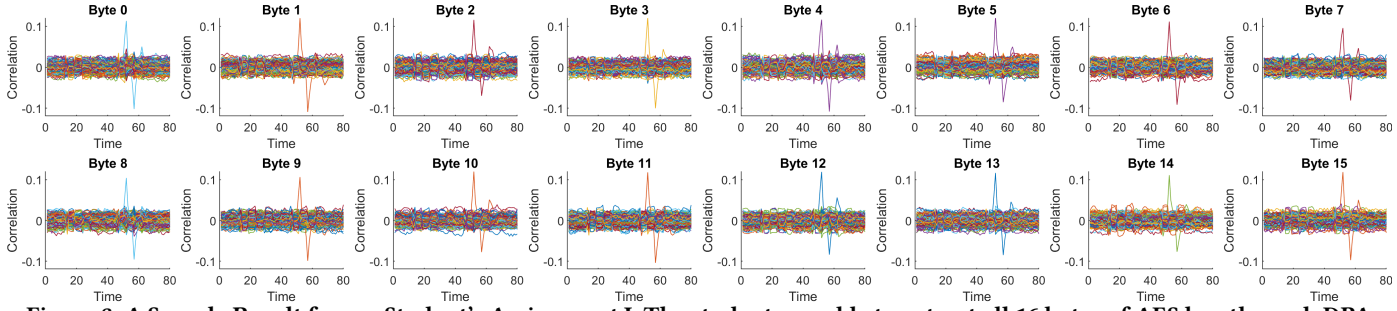

Figure 3: A Sample Result from a Student's Assignment I. The student was able to extract all 16 bytes of AES key through DPA

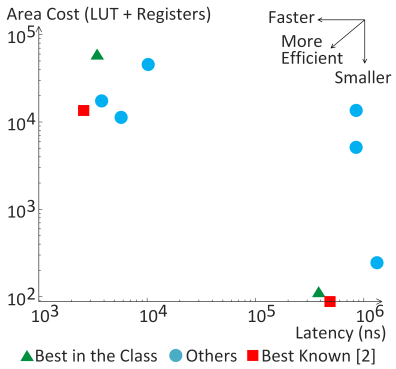

Figure 4: Results of Assignment II. The quality of the best implementations are close to the published work.

+ Registers) and the throughput (in Seconds per operation) of those designs and compares it with the best published prior work [2]. The student's designs vary in quality but yield a sufficient coverage of the design space. The results show that some graduate students can achieve almost an expert-level design that is close to the published work—the fastest design is only 25% slower than the previous best and the smallest design is only 27% larger. Given more design time, these designs can arguably reach the same level of optimality.

4.3 Assignment III: Implementation Attacks on Post-Quantum Encryption

The purpose of the last assignment is to critically analyze a cryptographic hardware and devise novel implementation attacks. This assignment asks to deploy the post-quantum encryption developed during Assignment II into a real FPGA (on SAKURA-G) and apply implementation attacks to recover the secret key. We want to spark the creativity of the students and let them explore different attacks. Prior to the assignment, the instructor and paper presentations extensively covered fault attacks and such attacks are further incentivized through bonus points. This assignment is organized with groups of 4 and is further broken down into two steps.

4.3.1 III-A: SAKURA-G Integration and Verification. During the first step, the students are asked to verify their design running on the FPGA platform. We deliberately design the testbench of Assignment II to match it with the I/O system interface of SAKURA-G. Despite this enforced compatibility and verified simulations, several groups had difficulty integrating their hardware mainly due to wrong interface assumptions and Synthesis/P&R issues. Having groups of 4 allowed students to have two options regarding which design to pursue. We also provided a GUI to control the hardware execution through software: to set or randomly generate inputs and keys, control the number of inputs/measurements, and verify the result in real-time by comparing it to a software emulation.

4.3.2 III-B: Attacks on the Developed Hardware. All four groups eventually have gone after SPA vulnerabilities because they are indeed simple to carry out. One group explored applying fault attacks but later pivoted due to fault automation challenges. Figure 5 demonstrate the vulnerability: when the secret key bit is 1, there is considerably more actions during lattice polynomial multiplication (partial product calculation and intermediate sum update) resulting in higher switching activity. Therefore, the secret key can simply be read on the power trace—a post-quantum secure encryption scheme turns out to be surprisingly vulnerable to implementation attacks and this attack can be deduced/conducted by graduate students who had no knowledge of the field two months ago!

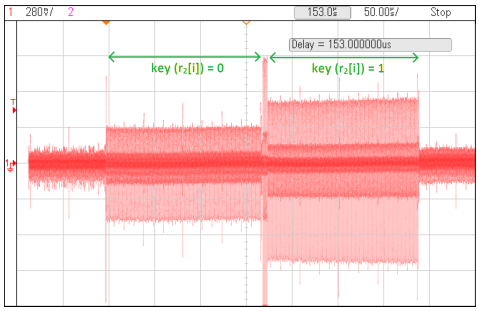

Figure 5: A Result from Assignment III. The group was able to break the post-quantum cryptosystem through SPA.

An interesting modification to this assignment could be organizing it in a capture-the-flag style where groups work on breaking each other's implementations—this can be further augmented by making students develop side-channel defenses. We can also request carrying out more attacks to exhaust the SPA option and to inspire groups to investigate other side-channel and fault attacks.

4.4 Calibrating Assignment Difficulty

The time requirement for each assignment will be a distribution as there will always be some that will progress faster than the others. During the last lecture, we conducted a written survey and ask students the time they spend on course components. We obtained 15 completed, anonymous responses. The average time spent on Assignment I, II, and III were respectively 13.5, 22.7, and 19.2 hours. Meanwhile, it was 13.3 and 3.8 hours respectively on paper presentation and review. Due to the gap between the last lecture and the final project (see Table 1), we could not get this number for final projects. These numbers may be biased because students that spend more time may already have dropped or withdrawn from the course, but they are consistent: we asked the time spent on Assignment II reports to cross-validate, which revealed an average of 22.9 hours, which is very close to the 22.7 hours obtained in the survey. A result we derive from these numbers is to extend or add another assignment after the first one to balance assignment load.

5 COURSE OUTCOME AND ANALYSIS

5.1 Final Research Projects

Table 2 presents the final projects pursued by the students ranked in terms of success. 3 projects have accomplished the proposed work, 5 had partial success completing pieces of the project, and 3 have failed to produce any reasonable deliverable. An interesting result of this course component is that all groups, with some guidance, were able to come up with a novel project that can improve the state-of-the-art. The three successful projects can clearly be turned into a research paper with some effort—the first one is a solid work on its own and the other two can be combined to this end. It is not surprising that the best project is done by the same student that achieved the best result in Assignment II. This student used his background on GPU architectures with the topics learned in the class to realize a low overhead side-channel defense for GPU timing side-channels. We emphasize again that these are students had little to no experience on hardware security yet they can be raised to a level not just to comprehend the existing work but to contribute to it. Indeed, we are following up on several of these projects with the goal of submitting them as a research paper during Spring 2019.

5.2 Course Participation and Interest

The capacity of the course was initially set to 30 students but this limit was raised to 45 due to increased demand. After the first

Table 3: Course Evaluation Results

#	Question	Grades Given					Course Statistics				Department Statistics			
		5	4	3	2	1	**Mean**	Std	SEM	Nr	**Mean**	Std	SEM	Nr
1	The instructors teaching aligned with the courses learning objectives/outcomes	8	3	0	2	0	**4.3**	1.1	0.31	13	**4.3**	0.9	0.03	663
2	The instructor was receptive to students outside the classroom	10	3	0	0	0	**4.8**	0.4	0.12	13	**4.4**	1	0.04	656
3	The instructor explained material well	4	6	1	2	0	**3.9**	1	0.29	13	**4.1**	1.1	0.04	660
4	The instructor was enthusiastic about teaching the course	10	3	0	0	0	**4.8**	0.4	0.12	13	**4.4**	0.9	0.03	664
5	The instructor was prepared for class	8	4	1	0	0	**4.5**	0.7	0.18	13	**4.4**	0.9	0.03	665
6	The instructor gave useful feedback.	7	5	1	0	0	**4.5**	0.7	0.18	13	**4.2**	1.1	0.04	658
7	The instructor consistently treated students with respect	10	3	0	0	0	**4.8**	0.4	0.12	13	**4.5**	0.9	0.03	668
8	Overall, the instructor was an effective teacher	6	3	2	2	0	**4**	1.2	0.32	13	**4.2**	1.1	0.04	660
9	The course materials were valuable aids to learning	6	4	1	2	0	**4.1**	1.1	0.31	13	**4.2**	1	0.04	666
10	The course assignments were valuable aids to learning	8	3	1	1	0	**4.4**	1	0.27	13	**4.2**	1.1	0.04	656
11	This course improved my knowledge of the subject	8	3	0	1	1	**4.2**	1.3	0.36	13	**4.3**	1	0.04	664
12	Overall, this course was excellent	8	3	0	1	1	**4.2**	1.3	0.36	13	**4.1**	1.1	0.04	663

Std=Standard Deviation, SEM=Standard Error of Mean, Nr=Number of Responses

Table 2: Final Research Projects Ranked in Terms of Success

#	Final Project Topic	Size	Results
1	Masking-inspired side-channel defenses for GPUs	1	Success
2	High-level synthesis of R-LWE decryption	2	Success
3	High-level synthesis of SIMON block cipher	2	Success
4	Enhanced power monitors for remote side-channel attack	2	Partial
5	Novel side-channel attacks on R-LWE encryption	1	Partial
6	Side-channel driven design-space exploration	2	Partial
7	High-level synthesis of PRESENT block cipher	2	Partial
8	Hardware realization of BIKE post-quantum scheme	2	Partial
9	Accelerating memory-hard functions on FPGAs	2	Failure
10	Hardware realization of LOTUS post-quantum scheme	1	Failure
11	Compiler optimizations for lightweight cryptography	1	Failure

week, 42 students were enrolled in the class. This number, however, gradually reduced to 23 after the drop period and withdrawals. 18 out of the 23 enrollments were taking the course with full credits. The average number of enrollment for special topics courses in the department is 13 and our course is ranked 4th out of 14 courses offered during the semester in terms of graduate enrollment—this clearly shows the interest of students to the topic. 5 out of this 23 participants were taking the course as an audit with pass/fail option. This number is significantly high compared to 4 other audits (out of 180) in other special topic courses, which further implies student interest but inability to take the course due to the other coursework.

5.3 Course Evaluations

Table 3 summarizes course evaluation scores. Overall, the course is on par and even slightly above average compared to department's statistics. The students rated instructor's feedback, availability, and enthusiasm relatively higher, while grading the teaching effectiveness lower—being instructor's first course, this score is expected to increase in the future iterations. The written comments (reducted for anonymity) can be accessed using the following link: https://www.dropbox.com/s/v8ixbgyws9xlb63/written-comments_Redacted.pdf?dl=0. We further discuss the feedback in Section 6 as they relate to future changes of the course. In addition to the formal evaluations, we also asked students to comment on their likes/dislikes about the course during the last lecture's survey. As the main trend, out of the 15 reported, 9 commented they liked hands-on assignments and 5 stated they disliked (the lack of) the level of depth.

5.4 Course Grade Distribution

Table 4 summarizes student's course component scores and the final letter grades. The majority of the participants have succeeded in the assignments, presentations and evaluations thus the deciding factor in the grades has been the final research projects—those who succeeded (or partially succeeded) got significantly better grades. The median letter grade of the class is A-, while the mean is B+ due to a few very low scores. The distribution of A, B, and C grades are respectively 56%, 28%, 16%, which is comparable to the department's Fall 2017 special topics course distribution of 66%, 29%, 5%.

6 COURSE CHALLENGES

To help adoption of a similar course, this section discusses the major challenges, potential resolutions, and our future plans.

6.1 Participant Demographics

The department's Master of Science in Electrical and Computer Engineering attracts a large number graduate students. This program has a non-thesis option, which is an intensive, three-semester program with a heavy course load. The majority of the course participants belonged to this profile: 16 out of 18 (22 out of 25 counting audits) were in this category, whereas the other 3 students were 1 senior undergrad and 2 PhD candidates. While the students performed competently when given clear tasks with step-by-step instructions, the challenge has been to enforce out-of-the-box thinking with the assignments (which is arguably necessary for cybersecurity) and to impose a research-oriented mindset for the final project. This is reflected in assignment results, where they spent considerably less time in the first assignment, in final projects that failed, and in course evaluations where several students raised concerns about research aspects. The heavy coursework also limits the amount of time this group of students can spend on each course. It would be an interesting study to teach this course exclusively to PhD students *with* master's thesis (i.e., students with some research background and relatively lower course load) and compare their results/reactions with the current group.

6.2 First-Time and One-Time Challenges

A major challenge of the course has been finding the right infrastructure and preparing the basic setup to teach hardware security. We have used the SAKURA-G board [7], which is a platform designed especially for side-channel attacks, fault injection attacks, physical unclonable functions and dynamic reconfiguration. This selection enabled us to prepare the assignments with relatively less effort. We did not provide a full automation for the side-channel and fault attacks that can integrate the PC, oscilloscope, voltage supply source, and SAKURA-G, but instead asked students to prepare it for Assignment III-B. Although the automation was successfully developed for the scope of that assignment, it was not sufficient for one final project and caused its failure.

One particular challenge was to obtain a sufficient number of boards from the SAKURA-G providers in Japan in short notice—we were able to initially get 3 boards in three months and 2 more boards afterwards, which led Assignment III to be organized in groups of 4 due to its heavy reliance on measurements. Performing this assignment with groups of 2 (with 12 boards) would have been a better option but would cost an extra $10,500 USD given the $1500 board price. Another option is using the Hardware Hacking Security Education Platform (HaHa) [19] once it becomes commercially available. There is a clear need for cheap, easily-accessible boards sold in US with more extensive hardware/software infrastructure.

Table 4: Score and Letter Grade Statistics of Participants

Component	Max.	Min.	Median	Mean	Std	SEM
Assignment I	10	0	10	9.04	2.86	0.12
Assignment II	10	0	10	8.57	2.88	0.13
Assignment III-A	5	0	5	4.78	1.04	0.05
Assignment III-B	5	0	5	4.65	1.04	0.05
Paper Presentations	20	16	18	18.26	1.28	0.07
Paper & Presentation Review	10	0	9.75	8.81	2.37	0.13
Project Proposal	15	11	15	13.11	1.32	0.07
Project Results	25	5	20	18.28	5.84	0.32
Total Socre	**100**	**71.8**	**90.15**	**87.39**	**8**	**0.44**
Letter Grade	**A+**	**C+**	**A-**	**B+**		

Std=Standard Deviation, SEM=Standard Error of Mean

6.3 Breadth vs. Depth

It is particularly challenging to balance the breadth vs. depth of course topics. On the one hand, introducing new concepts is crucial to make the future system designers aware of common hardware security issues. On the other hand, the depth of knowledge enables working on related concepts after the course. The current emphasis seems to be too much on the breadth—6 out of 8 written evaluation responses comment focusing more on depth rather than breadth. This trend is also visible in our last lecture's survey. One option to resolve this issue is to separate this course into two courses or to set a cryptography course as prerequisite. We do not intend to follow this direction. Another option is to remove some content, e.g., architectural aspects to cover more depth—this may have drawbacks because some students may exclusively be interested in those concepts. If course enrollment grows, it may not be possible to have the paper presentation component so they can be replaced for a more efficient coverage of depth through regular lectures.

6.4 Drop/Withdraw/Audit Related Issues

The course's reliance on group work generates a layer of complexity when students drop, withdraw, or change course participation to audit, which may happen without instructor's approval or notice. As a result, group sizes may change, papers may not be presented at the last minute, and the course schedule may alter accordingly. These cases have to be handled in an ad-hoc manner as there currently seems to be no systematic solution. This issue can be due to extended drop/withdraw periods, irregular audit procedures, and having tuition cap that leads students to take more courses early on and drop one later in the semester based on course load.

6.5 Planned Future Changes

Based on the lessons learned from the first iteration of this course, we plan to implement the following changes in the next edition.

- To improve it's understanding, we plan to add an assignment on fault attacks towards lattice-based post-quantum (R-LWE) cryptosystems. This can be done after Assignment II.
- We plan to extend Assignment III by asking a defense implementation and more advanced attacks beyond SPA.
- We will reconsider paper presentation component of the course. Due to scheduling challenges and scalability issues, we may replace it with regular lectures on paper's topics.
- We aim to start final projects earlier in the semester and ask for 1 or 2 progress reports. The caveat of this approach is that earlier project proposals can limit the scope since the students have not yet been exposed to several topics.
- We will explore alternatives for the development platform.
- We plan to include several short quizzes to motivate students to further study the course contents in detail.

7 CONCLUSIONS AND FUTURE DIRECTIONS

There is a critical need to raise the next-generation of hardware engineers with the awareness of hardware security. To that end, we discussed the curriculum design, assignment development, course organization, and associated challenges of a new course with specific focus on cryptographic engineering for post-quantum systems. We provide a basis for improvement and give recommendations for others who wish to offer a similar course. We strongly emphasize the importance of public/formal discussions, peer-reviewed publications, and collaborations on better establishing such courses. Although we do not discuss it in the scope of this work, a more scalable approach is an online course on hardware security [13], which would definitely have its own unique challenges. Another issue is the lack of a textbook to teach the course.

8 ACKNOWLEDGEMENTS

This work was supported in part by the NSF under Grants No. 1850373. Titan Xp GPU used is donated by Nvidia Corporation.

REFERENCES

[1] Miklós Ajtai. 1996. Generating hard instances of lattice problems. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*. ACM, 99–108.
[2] A. Aysu, M. Orshansky, and M. Tiwari. 2018. Binary Ring-LWE hardware with power side-channel countermeasures. In *2018 Design, Automation Test in Europe Conference Exhibition (DATE)*. 1253–1258.
[3] Dan Boneh, Richard A. DeMillo, and Richard J. Lipton. 2001. On the Importance of Eliminating Errors in Cryptographic Computations. *Journal of Cryptology* 14, 2 (01 Mar 2001), 101–119. https://doi.org/10.1007/s001450010016
[4] Johannes Buchmann, Florian Göpfert, Tim Güneysu, Tobias Oder, and Thomas Pöppelmann. 2016. High-Performance and Lightweight Lattice-Based Public-Key Encryption. In *Proceedings of the 2Nd ACM International Workshop on IoT Privacy, Trust, and Security (IoTPTS '16)*. ACM, New York, NY, USA, 2–9.
[5] O. Goldreich, D. Micciancio, S. Safra, and J.-P. Seifert. 1999. Approximating shortest lattice vectors is not harder than approximating closest lattice vectors. *Inform. Process. Lett.* 71, 2 (1999), 55 – 61.
[6] "Herjavec Group". 2017. *Cybersecurity Jobs Report – 2017 Edition.* Technical Report.
[7] H. Guntur, J. Ishii, and A. Satoh. 2014. Side-channel AttacK User Reference Architecture board SAKURA-G. In *2014 IEEE 3rd Global Conference on Consumer Electronics (GCCE)*. 271–274. https://doi.org/10.1109/GCCE.2014.7031104
[8] Yier Jin. visited on 2018-12-27. http://www.eecs.ucf.edu/~jinyier/courses/EEE4932/
[9] Jens-Peter Kaps. visited on 2018-12-27. https://ece.gmu.edu/~jkaps/teaching.html
[10] Paul Kocher, Joshua Jaffe, and Benjamin Jun. 1999. Differential Power Analysis. In *Advances in Cryptology — CRYPTO' 99*, Michael Wiener (Ed.). Springer Berlin Heidelberg, Berlin, Heidelberg, 388–397.
[11] Farinaz Koushanfar. visited on 2018-12-27. http://eceweb.ucsd.edu/~fkoushanfar/teaching/teaching.html
[12] Brian NeSmith. 2017. The Cybersecurity Talent Gap Is An Industry Crisis. https://www.forbes.com/sites/forbestechcouncil/2018/08/09/the-cybersecurity-talent-gap-is-an-industry-crisis/#3da7cbfca6b3
[13] Jim Plusquellic. visited on 2018-12-27. http://ece-research.unm.edu/jimp/HOST/index.html
[14] John Proos and Christof Zalka. 2003. Shor's Discrete Logarithm Quantum Algorithm for Elliptic Curves. *Quantum Info. Comput.* 3, 4 (July 2003), 317–344. http://dl.acm.org/citation.cfm?id=2011528.2011531
[15] Jeyavijayan Rajendran. visited on 2018-12-27. https://cesg.tamu.edu/all-courses/
[16] Jordan Robertson and Michael Riley. 2018. The Big Hack: How China Used a Tiny Chip to Infiltrate U.S. Companies.
[17] Patrick Schaumont. 2017. Secure Hardware Design - ECE 5520. http://rijndael.ece.vt.edu/schaum//teaching/5520/
[18] P. Shor. 1999. Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. *SIAM Rev.* 41, 2 (1999), 303–332. https://doi.org/10.1137/S0036144598347011
[19] Jason Vosatka Shuo Yang and Swarup Bhunia. 2018. Hardware Hacking Security Education Platform (HaHa SEP v2). In *2018 IEEE International Symposium on Hardware Oriented Security and Trust (HOST) – Hardware Demo.* http://www.hostsymposium.org/host2018/hwdemo/HOST_2017_hwdemo_8.pdf
[20] Mark Tehranipoor. visited on 2018-12-27. http://tehranipoor.ece.ufl.edu/hst.html
[21] I. Verbauwhede, D. Karaklajic, and J. Schmidt. 2011. The Fault Attack Jungle - A Classification Model to Guide You. In *2011 Workshop on Fault Diagnosis and Tolerance in Cryptography*. 3–8. https://doi.org/10.1109/FDTC.2011.13