Boston University College and Graduate School of Arts & Sciences

Undergraduate Academic Program Office
725 Commonwealth Avenue, Room 102

CAS/GRS New Course Proposal Form

This form is to be used when proposing a new CAS or GRS course.

This form should be submitted to Senior Academic Administrator Peter Law (617-353-7243) as a PDF file to pgl@bu.edu. For further information or assistance, contact Associate Dean Joseph Bizup (617-353-2409; jbizup@bu.edu) about CAS courses or Associate Dean Jeffrey Hughes (617-353-2690; hughes@bu.edu) about GRS courses.

DEPARTMENT OR PROGRAM: CAS Computer Science DATE SUBMITTED: 03/01/18

COURSE NUMBER: CAS CS 568

COURSE TITLE: Applied Cryptography

INSTRUCTOR(S): Mayank Varia

TO BE FIRST OFFERED: Sem./Year: __Spring__ /_2019_____

SHORT TITLE: The "short title" appears in the course inventory, on the Link University Class Schedule, and on student transcripts and must be 15 characters maximum *including spaces*. It should be as clear as possible.

A	P	P	L	I	E	D		C	R	Y	P	T	O	

COURSE DESCRIPTION: This is the description that appears in the CAS and/or GRS Bulletin and The Link. It is the first guide that students have as to what the course is about. The description can contain no more than 40 words.

Introduces the science and art behind the design, security analysis, and implementation of modern day cryptosystems that protect privacy and authenticity of data in transit and at rest. Demonstrates how cryptosystems evolved to withstand systems-level threats and mathematical cryptanalysis.

PREREQUISITES: Indicate "None" or list all elements of the prerequisites, clearly indicating "AND" or "OR" where appropriate. Here are three examples: "Junior standing or CAS ZN300 or consent of instructor"; "CAS ZN108 and CAS ZN203 and CAS PQ206; or consent of instructor"; "For SED students only."

1. State the prerequisites:

Requirements: CAS CS 210 along with mathematical maturity at the level of CAS CS 235 and 237, or consent of instructor.

2. Explain the need for these prerequisites:

This course combines ideas from algorithms, complexity theory, electrical engineering, and mathematics (probability and algebra). The prerequisites listed above are meant to ensure general familiarity with computer systems and math, but without being so onerous as to be overly restrictive. The class is specifically structured to account for the fact that students are unlikely to have significant experience in all

4 categories listed above. As such, lectures do expose students to the topics within these disciplines that are relevant to the course, and the out-of-class effort factors in the time required for students to learn about topics with which they are less familiar.

CREDITS: (check one)

☐ Half course: 2 credits ☐ Variable: Please describe.

X Full course: 4 credits ☐ Other: Please describe.

Provide a rationale for this number of credits, bearing in mind that for a CAS or GRS course to carry 4 credits, 1) it must normally be scheduled to meet at least 150 minutes/week, AND 2) combined instruction and assignments, as detailed in the attached course syllabus, must anticipate at least 12 total hours/week of student effort to achieve course objectives.

Previous iterations of this course (offered as CAS CS 591 V1 in the Spring 2016, 2017, and 2018 terms) have met for 150 minutes per week. Out of class weekly effort is anticipated to be the following: 5-8 hours on weekly lab assignments, 2-4 hours per week on required textbook reading, and 2-4 hours per week to persue relevant portions of the optional reading assignments based upon the student's own strengths and weaknesses (cf. the justification for prerequisites above).

DIVISIONAL STUDIES CREDIT: Is this course intended to fulfill Divisional Studies requirements?

X No.

☐ Yes. If yes, please indicate which division _____ and explain why the course should qualify for Divisional Studies credit. Refer to criteria listed here and specify whether this course is intended for "short" or "expanded" divisional list.

HOW FREQUENTLY WILL THE COURSE BE OFFERED?

☐ Every semester ☐ Once a year, fall **x** Once a year, spring ☐ Every other year

☐ Other: Explain:

NEED FOR THE COURSE: Explain the need for the course *and* its intended impact. How will it strengthen your overall curriculum? Will it be required or fulfill a requirement for degrees/majors/minors offered by your department/program or for degrees in other departments/school/colleges? Which students are most likely to be served by this course? How will it contribute to program learning outcomes for those students? If you see the course as being of "possible" or "likely" interest to students in another departments/program, please consult directly with colleagues in that unit. (You must *attach appropriate cognate comments using cognate comment form* if this course is intended to serve students in specific other programs. See FURTHER INFORMATION below about cognate comment.)

This course is based on three successful offerings at Boston University as CS 591 Special Topics, with enrollments growing in each offering. The course expands and adds a key element to the department's curriculum in the strategic area of Security and Cryptography. The department currently offers CS 538 Cryptography as the 'core' security course, which introduces students to the basic algorithms used to guarantee confidentiality and authenticity of data. The department also offers CS548 Advanced Cryptography and CS 558 Networking Security. These courses cover advanced topics and techniques that compliment and/or continue the content of CS 538. This proposed course will further expand our

crypto/security content by looking at the science, art, and actual application behind the design, security analysis, implementation, and cryptanalysis of modern day cryptosystems.

ENROLLMENT: How many undergraduate and/or graduate students do you expect to enroll in the initial offering of this course?

Extrapolating from prior offerings of this course, it is estimated that 35-50 students may enroll in the initial offering of this course, most of them are upperclass undergraduates.

CROSS-LISTING: Is this course to be cross-listed or taught with another course? If so, specify. Chairs/directors of all cross-listing units must co-sign this proposal on the signature line below.

No.

OVERLAP:

1. Are there courses in the UIS Course Inventory (CC00) with the same number and/or title as this course?
   X No.
   ☐ Yes. If yes, any active course(s) with the same number or title as the proposed course will be phased out upon approval of this proposal.
   NOTE: A course number cannot be reused if a different course by that number has been offered in the past five years.
2. Relationship to other courses in your program or others: Is there any significant overlap between this course and others offered by your department/program or by others? (You must *attach appropriate cognate comments using cognate comment form* if this course might be perceived as overlapping with courses in another department/program. See FURTHER INFORMATION below.)

Yes, as noted above this class contains material that compliments and builds on material covered in CAS CS 538, 548, and 558.

FACILITIES AND EQUIPMENT: What, if any, are the new or special facilities or equipment needs of the course (e.g., laboratory, library, instructional technology, consumables)? Are currently available facilities, equipment, and other resources adequate for the proposed course? (NOTE: Approval of proposed course does *not* imply commitment to new resources to support the course on the part of CAS.)

None.

STAFFING: How will the staffing of this course, in terms of faculty and, where relevant, teaching fellows, affect staffing support for other courses? For example, are there other courses that will not be taught as often as now? Is the staffing of this course the result of recent or expected expansion of faculty? (NOTE: Approval of proposed course does *not* imply commitment to new resources to support the course on the part of CAS.)

Staffing required for this course is relatively small. It is expected that 1 Teaching Fellow and grader support proportional to enrollment will be needed to aid the instructor with grading of labs and exams.

BUDGET AND COST: What, if any, are the other new budgetary needs or implications related to the start-up or continued offering of this course? If start-up or continuation of the course will entail costs not already

discussed, identify them and how you expect to cover them. (NOTE: Approval of proposed course does *not* imply commitment to new resources to support the course on the part of CAS.)

None.

EXTERNAL PROGRAMS: If this course is being offered at an external program/campus, please provide a brief description of that program and attach a CV for the proposed instructor.

FURTHER INFORMATION THAT MUST BE ATTACHED IN ORDER FOR THIS PROPOSAL TO BE CONSIDERED:

- A complete week-by-week SYLLABUS with student learning objectives, readings, and assignments that reflects the specifications of the course described in this proposal; that is, appropriate level, credits, etc. (See guidelines on "Writing a Syllabus" on the Center for Teaching & Learning website.) Be sure that syllabus includes your expectations for academic honesty, with URL for pertinent undergraduate or GRS academic conduct code(s).

- Cognate comment from chairs or directors of relevant departments and/or programs. Use the form here under "Curriculum Review & Modification." You can consult with Joseph Bizup (CAS) or Jeffrey Hughes (GRS) to determine which departments or programs inside and outside of CAS would be appropriate.

DEPARTMENT CONTACT NAME AND POSITION**: Mayank Varia, Research Associate Professor**

DEPARTMENT CONTACT EMAIL AND PHONE: varia@bu.edu, 617-353-8919

DEPARTMENT APPROVAL: _____ 3/1/18

Department Chair — Date

_____

Other Department Chair(s) (for cross-listed courses) — Date

SYLLABUS FOR APPLIED CRYPTOGRAPHY: DESIGN & PRACTICE
BOSTON UNIVERSITY CAS CS 591 V1, SPRING 2017

MAYANK VARIA
VERSION 1.0 – JANUARY 23, 2017

1. Course information

1.1. **Course description.** This course introduces the science and art behind the design, security analysis, implementation, and cryptanalysis of modern day cryptosystems.

First, we will examine several primitives including block ciphers and collision-resistant hash functions, which we will apply in order to design cryptosystems that protect the privacy and authenticity of data at rest and in transit. Second, we will examine how cryptography can overcome, or be harmed by, systems security concerns. Third, we will explore the state of the art in secure messaging systems that leverage public and secret key cryptography to protect communications even in the case of prior or future device compromise. Finally, we will examine the mathematical strength of block ciphers and hash functions toward common types of mathematical cryptanalysis. The course Piazza page has a detailed schedule that will be kept up to date throughout the semester.

The course will have a hands-on approach, with many programming-based homework assignments and a final project. As a graduate-level topics course, it is expected that students not only submit regular homework assignments but also review relevant reading materials and engage in creative discussions outside of class.

This course is not currently part of the CS concentration in cryptography & data security[1]. If you are working toward the certification and would like credit, please send me a note.

1.2. **Prerequisites.** Students are expected to have familiarity with computer systems at the level of CS 210 and familiarity with algebra and probability at the level of CS 235 and 237, or equivalent. Consent of the instructor is required to take the course if you do not meet these pre-requisites. More advanced courses, including exposure to cryptography at the level of CS 538 or 558, are helpful but not required. Additional background in math and systems engineering may also prove handy.

I recognize that, as with any graduate-level topics course, each student comes into the course with a different set of background knowledge. It is expected that students will require some out-of-class time to learn about unfamiliar topics; the amount of coursework given is adjusted accordingly. If you need help on material that would be covered in prerequisite courses, feel free to ask your fellow students, post questions on Piazza, or come to my office hours.

1.3. **Lecture times.** This course meets Mondays and Wednesdays from 12:20pm to 1:35pm in MCS (111 Cummington Mall) room B21. Registered students are expected to attend all lectures.

1.4. **Instructor.** The instructor is Dr. Mayank Varia. My office is located at MCS (111 Cummington St) room 164. My email address is varia@bu.edu. Please only use email for personal or administrative matters. If you submit a technical question via email (even one that should only be sent to me and not the rest of the class), I will either ignore it or ask you to re-post the question on Piazza (see next section) with the appropriate privacy protections set.

[1] http://www.bu.edu/cs/undergraduate/computer-science-major/concentration-in-cryptography-and-data-security/

1.5. **Course websites.** There are two websites that you will need to use during this class. Please **sign up for both** asap!

The course's Piazza page is located at: `https://piazza.com/bu/spring2017/cascs591v1/home`. Please sign up to the Piazza page with an email address that you will actually check. Official course announcements will regularly be sent via Piazza; it is your responsibility to read them. I strongly encourage you to use Piazza so that you may continue learning outside of lecture.

Second, we will use SageMathCloud for all problem sets in this course (see Section 2.1). Please **sign up for a SageMathCloud account** asap at `https://cloud.sagemath.com` using your `@bu.edu` email address so that I can match up your Sage account with your student registration information. Once your SMC account is connected to the course, you will need to pay $9 to upgrade your account to one with web access and access to faster machines; if this payment presents a hardship to you, let me know. You must upgrade your account by **Wednesday, February 1**.

Sage is a Python-based computer algebra system that is capable of doing many things, although we will only use a small fraction of its power; it also has extensive web-based and in-use documentation available; see the Piazza resources page for links.

1.6. **Office hours.** I plan to hold weekly office hours on Tuesdays from noon to 2pm and on Fridays from 3pm to 5pm. If I need to move the office hours, I will announce alternate times in class and on Piazza. If you want to meet with me but cannot make office hours, please send me a private note on Piazza with at least 3 suggestions for times that you are available, and we will work to find a meeting time.

2. Assignments

This course will contain three types of assignments: problem sets, tests, and a final project. All assignments must be completed in accordance with the collaboration policy specified in Section 3.4.

2.1. **Problem sets.** This course will have bi-weekly problem sets. Problem sets may contain problems that require written solutions as well as computer-based problems that require you to submit code that runs using the Python-based computer algebra system Sage.

All problem sets will be posted to SageMathCloud at least 1 week prior to their due date and announced during lecture. You can work on the assignments directly on the web; remember to **save your work** by clicking on the save icon (the system will auto-save your work every minute or so, but better to be safe than sorry). At the due date/time, assignments will automatically be collected for grading. Typically, the problem sets shall be due on a **Thursday evening at 9pm**, unless posted otherwise on the problem set itself. Failure to view Piazza and SageMathCloud will **not** be considered as a valid excuse for incomplete homework.

2.2. **Reading.** Each week I will post a paper or video to Piazza for reading/viewing. Typically, reading should be completed **before the Monday lecture**. Feel free to discuss the reading assignments on Piazza; additionally, each week I will create a special Piazza thread in which you should post 1-2 sentences stating either (1) an insight you gained from the paper or (2) a question you had when reading the paper. Note that you should not merely summarize the paper; the point is to share with your classmates something that came into your head as you were reading. You don't need to post every week, but you should be posting at least 60% of the time or so, with higher participation rates being rewarded in grading (see Section 3.2). We will use the posts as a starting point for discussion about the paper at the beginning of lecture.

2.3. **Test.** There will be one in-class test in this course, held on **Monday, April 3**. Please reserve this date on your calendar now. If you have a valid conflict, let me know about it as soon as possible. This course will not have a final exam.

2.4. **Project.** This course includes a final project, which requires the submission of a written report and the delivery of a presentation in class. You may work in groups of 2 or 3 people (larger or smaller group sizes are discouraged, but may be allowed with advance approval from the instructor). The amount of work is expected to scale linearly with the number of people in the group.

Students who are auditing the course are welcome to participate in the final project. However, mixing registered and non-registered students in a final project requires approval from the instructor; this is mostly to ensure that the registered students agree that they are responsible for the project whether or not the non-registered students continue to participate.

The scope of acceptable projects is broad: I encourage you to think about potential topics early and to post thoughts on Piazza and/or visit office hours so that I may help to refine your thoughts. In general, projects are likely to fall into one of these broad categories.

(1) Design of a new cryptographic cipher or primitive, at the mathematical or algorithmic level. It is very likely that this design will not be "from scratch" but rather an improvement over an existing design, perhaps inspired by original or prior cryptanalysis.
(2) Implementation of an existing cipher/primitive or library consisting of several primitives that offers some benefit offered over prior work, such as usage in a different programming language/environment, improved resilience to side channel attacks, etc.
(3) Extension of an existing crypto protocol or library to include (some combination of) additional user functionality, performance, or security benefits.

3. Evaluation

3.1. **Rubric.** Your grade in this course will be determined as follows: 50% problem sets, 20% test, 20% final project, and 10% reading and participation.

3.2. **Participation.** Participation during lecture and on Piazza will factor into decisions about borderline grades: the instructor may adjust grades upward for intelligent participation during lectures or downward for substantially missing lecture time.

Remember, the entire purpose of a topics course is to learn about new, cutting-edge material that has not yet been painstakingly recorded into a single textbook and perfected through decades of lecturing. Often, your best source of knowledge will be each other, not me. So, please use Piazza to continue the dialog between lectures! If you simply attend the lectures and do the homework without engaging with your fellow students, I think you will have lost a valuable learning opportunity.

On the flip side, any behavior during class that disrupts other students' ability to learn will not be tolerated; you will be asked to leave the classroom.

3.3. **Conduct code.** All students are responsible for adhering to the BU academic conduct code (`https://www.bu.edu/academics/policies/academic-conduct-code/`) at all times. Additionally, you must adhere to the collaboration policies stated in 3.4 for each assignment type; failure to adhere to the collaboration policies will result in a grade of 0 for the assignment and may be referred to BU's Academic Conduct Committee. Finally, this course uses a wiki page to facilitate sharing of project work; however, defacing or deleting your classmates' work on the wiki will not be tolerated. Intentionally misrepresenting the work of others may result in a failing grade for the course.

3.4. **Collaboration policy.** The course's collaboration policy is an attempt to balance the worthwhile exchange of ideas between students with the importance of providing fair grades to all. Please read the rules below carefully. If you are uncertain whether a particular kind of interaction with someone else violates the rules listed below, please **ask me beforehand**. If you cannot reach me, at the very least include a description of your interaction in your submission. If you acknowledge something that I deem to be a violation of the rules, I may lower your grade but I will not consider

the matter cheating. Unacknowledged violations that I discover will be referred to BU's Academic Conduct Committee, which can recommend punishments as harsh as suspension or expulsion.

Collaboration rules for problem sets are fairly liberal. In general, you are encouraged to discuss fundamental concepts (e.g., the material covered in lecture) with other students outside of class, either in person or via Piazza. Additionally, you are welcome to work with classmates to discuss the problems and their possible solutions together. There are three rules that you should keep in mind while working on the problem sets:

(1) While you may discuss the problems with your classmates, all of the actual writing/code submitted must be **entirely your own work**. Copying another student's work, or allowing your work to be copied, will result in a grade of 0 on the assignment and possible further action.
(2) At the beginning of the response to each problem (e.g., in a comment block for coding assignments), you must **list all classmates** with whom you've discussed the problem.
(3) You may not simply search the Internet for answers. The only permitted internet resources are the textbooks/papers linked on the Piazza course schedule page and any websites mentioned in the problem set itself.

Just to repeat: *failing to list collaborators in problem set solutions, or looking at written solutions of fellow students before submitting your own work will be considered cheating.*

The final project has an even more liberal policy: you are welcome and encouraged to discuss your work with your classmates (even students outside of your group), seek arbitrary publications or other resources on the web, etc. You may even share code with others (e.g., to incorporate another project's ideas into your own work). In fact, I encourage you to post your code on public forums like GitHub. There are only two rules to remember here:

(1) Your submitted work must cite **all** of the sources you consult, including personal conversations with people outside of your project! Be verbose: if you are uncertain whether an interaction is citation-worthy, please include it. Violations of this policy will be treated as plagiarism and referred to BU's Academic Conduct Committee.
(2) You will be graded on your novel contributions. So, while I encourage you to build upon the work of others, make sure that your additional contribution is a substantial one!

The test is intended to reflect individual ability; working with others or using any electronic resource is not allowed. It will be stated in advance whether notes are permitted during each test.

3.5. **Late policy.** An assignment is considered late if it is not submitted using the appropriate method by the stated time. So that I may post solution sets as soon as possible, late homework will not be accepted. On the plus side, the lowest 2 assignments will be dropped from your grade; you may use this flexibility however you see fit. Each problem set will contribute relatively equal weight toward the final grade, though I may adjust the distribution slightly to give more weight to longer assignments.

Project presentations will not be accepted late without advance permission of the instructor, and tests will not be re-scheduled without advance permission of the instructor.

3.6. **Re-grading policy.** You have the right to request a re-grade of any problem set or project. Send me a private message through Piazza if you want an assignment to be re-graded. Please be aware though that I may then completely re-grade the entire assignment (not just the part in question), which may potentially result in a lower grade.

4. Resources

4.1. **Textbooks.** There are no required textbooks to purchase for this course, in part because the material we will cover is drawn from many sources. Fortunately, almost all of the resources we

will use are either freely available over the internet or can be downloaded using the BU library system's online subscriptions to Springer or IEEE. The Piazza page includes an extensive list of reference material that may be of help to you throughout the semester, as well as papers specific to the lecture material covered in each week's lectures.

Some of the textbooks and reading assignments require access to the websites of publishers like Springer, IEEE, and ACM. You will only be able to access these papers if you are on the BU network or if you VPN into it. Instructions to VPN into the BU network are located here: http://www.bu.edu/tech/services/support/remote/vpn/. Alternatively, prepending http://ezproxy.bu.edu/login?url= to the front of a URL allows you to view a single website through the BU network without the need to VPN.

4.2. **Software.** In addition to SageMathCloud itself, cryptographic libraries for Python like PyCrypto and cryptography.io will come in very handy during your problem sets. Additionally, there is a wide range of cryptographic libraries available for other programming languages that you are welcome to use in your project work (and in selected homework problems, if permitted explicitly in the problem statement). Java has built-in support for cryptography via the Java crypto extensions, C has hundreds of open-source crypto projects including the (in)famous OpenSSL (https://www.openssl.org/) and Mozilla NSS (https://developer.mozilla.org/en-US/docs/Mozilla/Projects/NSS), the Stanford Javascript Crypto Library (https://bitwiseshiftleft.github.io/sjcl/) is useful for web-based programming, Rust is beginning to receive support for advanced libraries such as Axolotl (https://github.com/wireapp/proteus), and so forth.

4.3. **Conferences.** As with most fields of computer science, cryptography is dominated by the conference model of publications, not the journal model. The following conferences accept dozens of papers per year in applied cryptography and therefore have a large body of material relevant to this course: Computer and Communications Security (CCS), Fast Software Encryption (FSE), IEEE Security & Privacy (Oakland), Real-world Crypto (RWC), Selected Areas in Cryptography (SAC), and USENIX Security. There are many other security venues that occasionally accept applied crypto papers.

To obtain papers from these conference, you should first try the Cryptography ePrint Archive at https://eprint.iacr.org, as most authors post their papers there. Otherwise, access to the conference proceedings is usually possible through SpringerLink (http://link.springer.com) or the BU library page.

4.4. **Seminars.** The Boston area is a bastion of cryptographic research, with arguably more resources available than any other city in the world. If you're interested in (or already!) pursuing postgraduate work in cryptography, I strongly encourage you to attend the following seminars: (1) BU's security seminar, held Wednesdays at 10am in the MCS building (https://www.bu.edu/cs/busec), and (2) MIT's cryptography and information security (CIS) seminars, held Fridays at 10:30am in the Stata Center (http://toc.csail.mit.edu/node/428). Signing up for the mailing lists on these websites is the best way to hear about these seminars. There are other Boston-area seminar series that occasionally cover crypto topics, such as those at Harvard or Microsoft Research.

Please note that almost all of the seminars will not be directly applicable to the material covered in class. If I do see a talk announcement for a relevant seminar, I will forward it to the class. Attendance will never be mandatory for any seminars held outside of class time.

4.5. **Websites.** In addition to our course website, there is extensive material on Wikipedia, Crypto StackExchange (http://crypto.stackexchange.com), social media, and personal blogs. Often times blog posts can be an excellent introduction to a new topic; however, in order to understand a topic in detail, a textbook or academic paper is almost always going to be a better resource.