CS 380/525: Introduction to Cryptography
Fall 2022
Class Project

This handout lists out some information about possible project topics and guidelines for the project. Projects are meant for individual work. Following are the milestones for the project; all submissions will be via Canvas:

1. *Project selection*: Tue., Sept. 29^{th}. You should look over the project topics, decide which one you have sufficient time for, and interest in, and get back to me with your project topic (an e-mail will do). I have included suggestions for possible project topics in this handout, but my list isn't an exhaustive list. If you'd like to explore something not on this list, but is relevant to our course topics, please feel free to ask me about it.

2. *Project Progress Report*: Thurs., Oct. 27^{th}. Progress report should be written in the template I've posted. By this time you should have collected enough references for your project, and thought through what goes in your project report. In case of a coding project, you should have the function prototypes and the skeleton of the program(s). In case of a paper-reading project, you should turn in a report (around 2 pages) that has a basic introduction to the topic you're investigating, a preliminary classification, and a list of all references you plan to use (around 5-7 references). In both cases, please make sure to format your references properly (see the other project handout for details).

3. *Complete project*: Due Thurs. Dec. 1^{st}, use the template I've posted.

1 Project Guidelines

There are two kinds of projects you can pick: a coding project, or a research paper-reading-based project.

1.1 Coding project

If you have not used it before, this is an opportunity for you to get to know the Charm cryptography library (see here `https://github.com/JHUISI/charm`, and here `http://charm-crypto.io/`). Charm is used for fast implementation of several crypto algorithms and protocols. It takes care of all the underlying math structures, so you don't have to define your own data types, etc. It supports all the structures in number theory we'll be talking about in the class. Some of the coding projects you can consider are:

1. Implement various public-key encryption and/or signature algorithms, e.g., RSA, ElGamal, Schnorr, DSA and compare performance.

2. Implement an advanced cryptographic algorithm, e.g., identity-based encryption or attribute-based encryption, and time its performance.

3. Implement the 3 modes of encryption we've covered in class, CBC, OFB and CTR, and compare their performance. You probably won't need the Charm library for this.

4. Implement a Merkle hash tree, and demonstrate how to do simple file integrity checking using it (might not need Charm).

1.2 Paper-reading project

This is meant to be a survey-based project. You are expected to read and classify current research in a given area of applied cryptography. The goal of the project is to get you familiar with the general flavor and tone of papers in applied cryptography, and to give you an introduction to current, exciting research topics that the security/privacy research community is currently working on. To that end, please do not summarize or describe a given technology, e.g., a description of how IPSec or Kerberos works, even though it "uses crypto" wouldn't be an acceptable topic.

Also, technical depth is important. The project isn't about merely collecting a bunch of papers, summarizing them in no particular order, and writing one-paragraph reviews on each of them (such an effort wouldn't fetch you more than 20% points in your project). The bulk of points is reserved for you to read through the papers you collected, and compare them w.r.t., approach taken, hardness assumptions, efficiency, scalability, trade-offs and compromises made, and other performance metrics.

1.2.1 Where to pick papers from?

Try picking papers from the top-ranked 4 security conferences: ACM CCS, NDSS, IEEE Security & Privacy ("Oakland"), and Usenix Security Symposium. Please make sure your papers are relatively recent – 2018 and after.

1.2.2 Project Topics

The following topics are merely suggestions. Feel free to pick a topic not on this list, but you'll need to run it past me before you start working on it. If the topic is of sufficient depth and relevant to this class, I'll probably be ok with it.

1. Bitcoin, cryptocurrencies, scalability issues, and/or misuse of them

2. Blockchain-enabled credit networks, payment networks

3. Other blockchain-enabled applications: accountability, surveillance, smart contracts and their efficiency issues

4. Attribute-based cryptography: either

   (a) Attribute-based encryption (ABE)

   (b) Attribute-based signatures (ABS)

5. Zero-knowledge proofs: either
   (a) Cut-and-choose interactive ZKPs
   (b) Non-interactive ZKPs (usually based on bilinear pairings)
6. Identity-based encryption (IBE)
7. Alternative signature schemes, e.g., group signatures
8. Anonymous communication
9. Verifiable encryption/fair exchange
10. Automated theorem provers/cryptographic proof-assistants