24
Course recap

It might be worthwhile to recall what we learned in this course:

- Perhaps first and foremost, that it is possible to *mathematically define* what it means for a cryptographic scheme to be secure. In the cases we studied, such a definition could always be described as a "security game". That is, we first define what it means for a scheme to be *insecure*. Then, a scheme is secure if it is not insecure. The notion of "insecurity" is that there exists some adversarial strategy that succeeds with higher probability than what it should have. We normally don't limit the *strategy* of the adversary but only his or her *capabilities*: its computational power and the type of access it has to the system (e.g., chosen plaintext, chosen ciphertext, etc.). We also talked how the notion of *secrecy* requires *randomness* and how many real-life failures of cryptosystems amount to faulty assumptions on the sources of randomness.

- We saw the importance of being *conservative* in security definitions. For example, how despite the fact that the notion of chosen ciphertext attack (CCA) security seems too strong to capture any realistic scenario (e.g., when do we let an adversary play with a decryption box?), there are many natural cases where the using a CPA instead of a CCA secure encryption would lead to an attack on the overall protocol.

- We saw how we can prove security by *reductions*. Suppose we have a scheme S that achieves some security notion X (for example, S might be a function that achieves the security notion of being a pseudorandom generator) and we use it to build a scheme T that we want to achieve a security notion Y (for example, we want T to be a message authentication code). To prove T is secure, we show how we can transform an adversary B that wins against T in the security game of Y into an adversary A that wins against S in the security game of X. Typically, the adversary A will run B "in its belly", simulating for B the security game Y with respect to T. This can be somewhat confusing, so please re-read the last three sentences and make sure you understand this crucial notion.

- We also saw some of the concrete wonderful things we can do in cryptography:

- In the world of *private key cryptography*, we saw that based on the PRG conjecture we can get a CPA secure private key encryption (which in particular has key shorter than message), pseudorandom functions, message authentication codes, CCA secure encryption, commitment schemes, and even zero knowledge proofs for NP complete languages.

- We saw that assuming the existence of *collision resistant hash functions*, we can get message authentication codes (and digital signatures) where the key is shorter than the message. We talked about the heuristic of how we can model hash functions as a *random oracle* , and use that for "proofs of work" in the context of bitcoin and password derivation, as well as many other settings.

- We also discussed practical constructions of private key primitives such as the AES block ciphers, and how such block ciphers are modeled as pseudorandom permutations and how we can use them to get CPA or CCA secure encryption via various modes such as CBC or GCM. We also discussed the Merkle and Davis-Meyer length extension construction for hash functions, and how the Merkle tree construction can be used for secure storage.

- We saw the revolutionary notion of *public key encryption*, that two people can talk without having coordinated in advance. We saw constructions for this based on discrete log (e.g., the Diffie-Hellman protocol), factoring (e.g., the Rabin and RSA trapdoor permutations), and the *learning with errors* (LWE) problem. We saw the notion of digital signatures, and gave several different constructions. We saw how we can use digital signatures to create a "chain of trust" via certificates, and how the TLS protocol, which protects web traffic, works.

- We talked about some advances notions and in particular saw the construction of the surprising concept of a *fully homomorphic encryption (FHE)* scheme which has been rightly called by Bryan Hayes "one of the most amazing magic tricks in all of computer science". Using FHE and zero knowledge proofs, we can get multiparty secure computation, which basically means that in the setting of interactive protocols between several parties, we can establish a “virtual trusted third party” (or, as I prefer to call it, a “virtual Chuck Norris”).

- We also saw other variants of encryption such as *functional encryption*, *witness encryption* and *identity based encryption*, which allow for “selective leaking” of information. For functional encryption and witness encryption we don’t yet have clean constructions under standard assumptions but only under obfuscation, but we saw how we could get identity based encryption using the random oracle heuristic and the assumption of the difficulty of the discrete logarithm problem in a group that admits an efficient *pairing* operation.

- We talked about the notion of obfuscation, which can be thought as the one tool that, if exists, would imply all the others. We saw that virtual black box obfuscation does not exist, but there might exist a weaker notion known as “indistinguishability obfuscation” and we saw how it can be useful via the example of a witness encryption and a digital signature scheme. We mentioned (without proof) that it can also be used to obtain a functional encryption scheme.

- We talked about how quantum computing can change the landscape of cryptography, making lattice based constructions our main candidate for public key schemes.

- Finally, we discussed some of the ethical and policy issues that arise in the applications of cryptography, and what is the impact cryptography has now, or can have in the future, on society.

24.1 SOME THINGS WE DID NOT COVER

- We did not cover what is arguably the other “fundamental theorem of cryptography”, namely the equivalence of one-way functions and pseudorandom generators. A one-way function is an efficient map $F : \{0,1\}^* \rightarrow \{0,1\}^*$ that is hard to invert on a random input. That is, for any efficient algorithm A if A is given $y = F(x)$ for uniformly chosen $x \leftarrow_R \{0,1\}^n$, then the probability that A outputs x' with $F(x') = y$ is negligible. It can be shown that one-way functions are *minimal* in the sense that they are *necessary* for a great many cryptographic applications including pseudorandom generators and functions, encryption with key shorter than the message, hash functions, message authentication codes, and many more. (Most of these results are obtained via the work of Impagliazzo and Luby who showed that if one-way functions do not exist then there is a *universal posterior sampler* in the sense that for every probabilistic process F that maps x to y, there is an efficient algorithm that given y can sample x' from a distribution close to the posterior distribution of x conditioned on $F(x) = y$. This result is typically known as the equivalence of standard one-way functions and distributional one-way functions.) The fundamental result of Hastad, Impagliazzo, Levin and Luby is that one-way functions are also *sufficient* for much of private key cryptography since they imply the existence of pseudorandom generators.

- Related to this, although we mentioned this briefly, we did not go in depth into "Impagliazzo's Worlds" of algorithmica, heuristica, pessiland, minicrypt, cryptomania (and the new one of "obfustopia"). If this piques your curiosity, please read this 1995 survey.

- We did not go in detail into the design of private key cryptosystems such as the AES. Though we discussed modes of operation of block ciphers, we did not go into a full description of all modes that are used in practice. We also did not discuss cryptanalytic techniques such as linear and differential cryptanalysis. We also not discuss all technical issues that arise with length extension and padding of encryptions in practice.

- While we talked about bitcoin, the TLS protocol, two factor authentication systems, and some aspects of pretty good privacy, we restricted ourselves to abstractions of these systems and did not attempt a full "end to end" analysis of a complete system. I do hope you have learned the tools that you'd be able to understand the full operation of such a system if you need to.

- While we talked about Shor's algorithm, the algorithm people actually use today to factor numbers is the *number field sieve*. It and its predecessor, the quadratic sieve, are well worth studying. The (freely available online) book of Shoup is an excellent source not just for these algorithms but general algorithmic group/number theory.

- We talked about some attacks on practical systems, but there many other attacks that teach us important lessons, not just about these particular systems, but also about security and cryptography in general (as well some human tendencies to repeat certain types of mistakes).

24.2 WHAT I HOPE YOU LEARNED

I hope you got an appreciation for cryptography, and an understanding of how it can surprise you both in the amazing security properties it can deliver, as well in the subtle, but often devastating ways, that it can fail. Beyond cryptography, I hope you got out of this course the ability to think a little differently - to be paranoid enough to see the world from the point of view of an adversary, but also the lesson that sometimes if something sounds crazy but is not downright impossible it might just be feasible.

But if these philosophical ramblings don't speak to you, as long as you know the difference between CPA and CCA and I won't catch you reusing a one-time pad, you should be in good shape :)

I did not intend this course to teach you how to implement cryptographic algorithms, but I do hope that if you need to use cryptography at any point, you now have the skills to read up what's needed and be able to argue intelligently about the security of real-world systems. I also hope that you have now sufficient background to not be scared by the technical jargon and the abundance of adjectives in cryptography research papers and be able to read up on what you need to follow any paper that is interesting to you.

Mostly, I just hope you enjoyed this last term and felt like this course was a good use of your time. I certainly did.