Ning

25 September 2019

Computer Security

Class Summary

[OPENING]: Hashed out more explanations on private/public keys, and the TP section received more mathematical background.

[SUMMARY]: As a reminder, in a hash function, the ability to find collisions is the benchmark of whether or not your hash function is good or not.

[SUMMARY]: In order to guarantee security, a RNG that is able to securely (read: randomly) generate values is extremely important. Many systems fail in security due to bad RND performance for generating truly random keys. i.e: WPA2 has a flaw that makes the generation of keys to secure your wifi network not entirely random, meaning that these keys can be brute-forced in a reasonable amount of time. Randomness is important, and not trivial to emulate.

[SUMMARY]: When talking about symmetric encryption, the bright side is its speed (compared to asymmetric). Digital signature is the process of signing a document with the document and your private key, and verifying the signature with a public key. The example given here is that bitcoin is just a concept of verifying transactions with a public key and receiving a fee from the transaction. The digital signature process is deterministic.

[SUMMARY]: When talking about the real world transactions, we often use both asymmetric and symmetric encryptions.

[SUMMARY]: Public keys are public, but they need to be authentic. Meaning that we need to be able to tie them to the correct entity in order to guarantee security. This is done via certificates and a certificate process. (note: there is a MIT phD that once told me over dinner about his research, which is something along the lines of “verifying that the certificate given to you by the entity (in our case, google) is legit” and somewhere along the lines of “you send a mathematical proof to (google) and have them solve it, and send it back to you” or something like that. It was very interesting and if I ever remember his name, I’ll be sure to find more reading regarding such.)

[SUMMARY]: Group theory -> a group is a set of values with a certain operation that satisfies the conditions that the operation is commutative, there exists a neutral element (identity) as well as an inverse that results in the neutral element (identity). For instance, the group of integers with addition is a valid group. We then later on work through more examples and talk about Euclid’s Algorithm and modulos, in order to set up the notions required to understand RSA.