Group Members:

Kyle Connall

Tyler Montague

Zobe Murray

Jacob Simons

Mads Spink

Fady Youssef

Group 4: Symmetric Key Encryption (OTP or POTP)

Scenario 1:

This class examined the algorithms and math used to protect our personal information everyday. When information is transmitted over the internet it must be protected in order to prevent malicious actors from tampering with, or intercepting the data. This is done through algorithms that encrypt the data in order to prevent it from being read by someone "eavesdropping" on the internet. Using math, the data is encrypted making it look like complete gibberish. When this encrypted data is received by the intended recipient, it is decrypted to return it to its original form, the original message sent. Additionally, cryptography seeks to assure that the message sent is the same message that is received, meaning it was not tampered with. Using math, the recipient of the message is able to verify whether or not the message received is the same message that was sent. This class examined the math behind these algorithms and did thought experiments and "tests" to assure that the math both works, and assures security for the information.

This class is important to my education primarily because of my intentions to become a professional in the tech industry. The backbone of the modern infrastructure relies on digital communication, which as mentioned above, needs to be secure. Developing an awareness of what is going on behind the scenes when securing information is critical if you are going to be working on and in these systems- even if you are not a computer security professional. The knowledge of how these protocols work means that you can use them effectively or work within them effectively. Although it does go beyond just the practical application of the material. The skills you practice and develop while taking rigorous courses generally contribute to critical thinking and reasoning skills, which are integral to a successful professional and personal life in many instances.

Scenario 2:

Scenario response:

"Maybe if you gave me a million dollars?! Of course people, myself included, have the resources to learn to mine crypto that is worth some value, but if you're looking for a quick way to "get rich", mining bitcoins isn't the place to start. I've been taking this applied cryptography course that has taught me a lot of what it takes to secure sensitive data, including digital currency. This class was more on the topics of fundamental approaches to complex algorithms, and analysis but mining something like crypto takes a level of intuition and education in statistics, cryptography, and networking. Not to mention, it takes a costly amount of resources like hardware and lots of financial risk, it's just not worth it if you aren't sure of what you're doing nor have the resources to do it."

We are not using these fundamental ideas to mine bitcoin and "get rich", but we do use them to build our skills in analysis. I would say you can't just quickly "get rich" from mining bitcoin. Our goal is to explain to them how certain things get secure in our everyday life that we often take for granted. Certain aspects of the class relate to crypto mining such as securing transactions, maintaining data integrity, and identity verification, but the direct correlation between our class and "getting rich" was not well defined.

Scenario 3:

The content of this course will cover mathematical formulations and applications in various cryptographic concepts. This will include things such as proofs by reduction and applying formulas and definitions into security goals that make a protocol secure. In this course, you will also cover some ciphers, both old and new, including classic ciphers like Caesar cipher and Vigenere cipher, as well as more modern ones like One Time Pad. One Time Pad will go into more detail in the course when you start talking about pseudorandom one time pad, as you start covering material like cryptographic definitions and reductions. You will spend most of the semester for this course talking about these security protocols, their assumptions and their proofs. This class focuses on security, its definitions, assumptions about them, and maintaining integrity with these protocols and concepts. If you are looking to learn more about cryptocurrency then you are out of luck. For someone who likes math and would like to see how it can be applied into a cryptographic context, then this class will be very interesting to you. We also covered the importance of groups in public key encryption, and what properties make up a group. Probability theory is also an important concept in cryptography, so you will cover this pretty early on in the semester when taking the course. This class will especially test your analytical skills when covering some of the security protocols in the class, so if you have an interest in math and want to get into security, then this class may be a good choice for you. **As a side note, you will also cover the apocalypse and farming.**

Scenario 4:

Applied cryptography was an introduction into the realm of cryptography and its uses in cybersecurity, covering topics of fundamental approaches to complex algorithms, and analysis. It gave me an understanding of how to approach building and analyzing protocols for security, and how these protocols are being used in day to day data encryption and decryption. Concepts such as perfect security have become increasingly important in banking, communication, government, and other applications as the world continues to generate more data and bad actors improve their hacking and malware capabilities. Taking this course also allowed me to apply math concepts like groups, probability theory, and proofs through contradiction, to building secure models for sending and receiving data, which builds critical thinking and analysis skills.

Scenario 5:

One time pad (OTP) is an algorithm that is used to securely pass information from one individual to another. OTP is actually a set of two algorithms, encrypt and decrypt. Encrypt is used by the sender of the message to hide the actual message and cause it to be complete gibberish. The encryption algorithm takes a message and a key as input and outputs a ciphertext. The key needs to be the exact same amount of characters as the message. The encryption algorithm moves through the message and key, character by character and essentially combines the two. This is done by adding their ascii numbers together and then modding that number by the amount of allowable characters (27 if space is included as part of the alphabet). For example, if P is the first character of the message and X is the first character of the key, we would add 15 and 23, the respective ascii values of P and X, to get 38. Then 38 would be modded by 27 to get 11 which is the ascii value corresponding to L which would then be the first character of the ciphertext. This process is then repeated until the whole message has been encrypted. The recipient has access to that same key and receives the ciphertext. The recipient then inputs the ciphertext and key into the decrypt algorithm to receive the original message. The decrypt algorithm essentially does the opposite of the encrypt algorithm, subtracting the ascii value of the key from the ascii value of the ciphertext. Using the example above, 23 would be subtracted from 11 to get -12, if the resulting number is negative as in this case, 27 is added in order to reach the range of acceptable numbers (0 to 27 in this case). Adding 27 to -12 results in 15 which corresponds to the letter P, the first character of the original message. This process is repeated until the whole message is decrypted and the recipient is left with the original message. In the case of bit strings the key and the message to be encrypted might be xor-ed together, as xor is not lossy.

A pseudorandom function is a function that takes a bitstring of some length, *n,* and maps it onto a bitstring of some length *l*. A pseudorandom generator requires $n < l$, and therefore successfully “generating” a longer bitstring given a smaller bitstring as input. A pseudorandom function with $n < l$ the mapping also needs to be sufficiently difficult to crack in polynomial time. Because there are 2^n possible inputs, and 2^l possible outputs there are $(2^n)^{2\wedge l}$ possible mappings. Given sufficiently large numbers of *n* and *l* there are far too many permutations to be calculated and tested in polynomial time, and given that the mapping is generated with sufficient opaqueness, i.e. with some random value, and cannot be reverse engineered given a key.

Pseudo-random one time pad uses a Pseudo-random generator to map smaller bit strings of length *n* to length *l*, the length of the encrypted message. The output, *l*, is then used as the key in the one time pad function. This means that there can now be a key of length *n*, which can be much shorter than the length of the encrypted material, making it less computationally expensive to produce a key. This is secure as long as the PRG used is secure in polynomial time.

In classical cryptography OTP is the only encryption algorithm for ensuring perfect secrecy. Using OTP is a computationally intensive process for large amounts of information because the key must be as long as the message. Therefore, OTP is mostly used by governments and intelligence agencies when handling sensitive information and keys are exchanged in person.