Expanding Cybersecurity: The Ever-Evolving Information War

CS 4991 Capstone Report, 2022

Kalman Buterbaugh
Computer Science
The University of Virginia
School of Engineering and Applied Science
Charlottesville, Virginia USA
kb2cj@virginia.edu

Abstract

Encryption has always existed in some form to keep information private. Those wishing to access protected information have developed techniques to break the encryption. With every development in technology come improvements in encryption, along with code-breaking techniques that feed the battle for private information. In recent years, this war has found its home in the world of software. Technologies from private messaging to cryptocurrency seek to transfer information across networks securely, while criminals and spy agencies alike write malware in an attempt to steal data.

CS 4630: Defense Against the Dark Arts at the University of Virginia, covers standard malware techniques and common security practices designed to protect against them. I propose expanding that class to cover the effects that rapidly developing technologies such as AI, machine learning, and quantum computing have on both sides of the privacy battle. Spending a few weeks at the end of course exploring these ideas would help prepare students planning to enter the cybersecurity industry for the future of this rapidly changing field.

1. Introduction

The technical infrastructure of modern society is built on the assumption that secure communication is generally possible. We send emails, make electronic payments, use remote file storage, and do so much more based on the idea that our data is secure. Granted any sufficiently complex software will have security holes– this is the entire basis behind CS 4630. However, these security holes are called "vulnerabilities" precisely because they are flaws in a secure system. But what if a secure system was not possible at all?

The security of modern society is based on unbreakable cryptographic algorithms such as Transport Layer Security (TLS) and Secure Hash Algorithm (SHA). If technology is developed that is able to break these algorithms, the field of cybersecurity will need to adjust quickly to meet the new threat. It is therefore quite beneficial that we study emerging technological developments in this field now in order to be better prepared when they are fully realized.

2. Related Work

The need to send communication to a designated recipient without being intercepted by a third party is not a new problem. Singh identifies Herodotus as one of the first recorded examples of code writing in the fifth century BC (Singh, 1999, p.3). When one thinks of the field of cryptography, one often first thinks of simple substitution codes or transposition codes, which apply a simple to either switch each letter of a message for a different character or rearrange the characters according to a predetermined rule. A common example of this is the Caesar Cipher, which substitutes each letter for the letter three characters later in the alphabet. For example, "HELLO WORLD" would be encrypted as "KHOOR ZRUOG".

Simple codes like the Caesar cipher are a good introduction to cryptography, but long ago people discovered efficient algorithms for breaking them and so such ciphers are not useful for security today. Throughout time, people have developed more complex encryption techniques, such as the German Enigma machine used in World War II. These algorithms too have been broken by cryptanalysts (Wilcox, 2006).

In recent years, however, certain cryptographic algorithms have been written that are currently considered secure. Popular examples of these include the previously mentioned TLS, as well as the Elliptic Curve Digital Signature Algorithm (ECDSA). We see these modern ciphers everywhere in society: for example TLS is used by GMail and many other email services (Google, 2022) and ECDSA is the backbone behind the popular cryptocurrency, Bitcoin (Bitcoin, 2021). The central theme behind these algorithms is that decryption using the key can be done relatively quickly (polynomial runtime) but decryption without the key (i.e. breaking the cipher) requires exponential runtime.

As technology in Computer Science develops, our current cryptographic algorithms face the possibility of encountering a similar fate to former encryption techniques like the Caesar Cipher and the Enigma. One of the most prominent of these technologies is quantum computing, which has been shown to be able to break the popular Rivest-Shamir-Adleman (RSA) algorithm (Mermin, 2007) and could likely be used to break many other modern encryptions. Another emerging technology that is likely to have significant impact on both improving and breaking encryption is Artificial Intelligence (Brett et al., 2017).

3. Proposed Design

Given the amount of speculation necessary when studying technology that is still in development, the material to be added to the course will be higher-level and more conceptual, making it a good fit to be added to the final couple weeks of class.

The central focus of my proposed course modification is quantum computing, which would be the topic of most of the additional course content.

While understanding the particular science behind quantum computing is certainly beyond the scope of a single unit in a cybersecurity class, covering the general concepts behind a quantum computer will help students understand the general capabilities of quantum computers. Quantum computers are built on the principle of superposition so that each quantum bit (known as a “qubit”) is kept in a state of varying probability for being either 0 or 1. Algorithms for quantum computers are written in a way that attempts to manipulate this probability so the solution to the given problem is measured with a high probability (Aaronson, 2021).

In considering what problems can be solved by quantum computing and the effects that may have in cybersecurity, it is helpful to consider a brief discussion of time complexity classes. This will be familiar to students who have taken CS 3102: Theory of Computation. The topic is commonly framed around the famous P=NP debate, where P represents the class of problems that can be solved in polynomial time (i.e. “easy” problems) and NP is the broader class of all problems whose solutions can be checked in polynomial time. NP includes both polynomial time problems and problems that can only be solved in exponential time (i.e. “hard” problems).

Modern encryption is built on algorithms that can decrypt in polynomial time (class P), but attempting to decrypt without the key falls into the NP class outside of polynomial time. The class of problems solvable on a quantum computer is known as BQP. Aaronson (2010) gives an analysis of BQP as it relates to P and NP, arguing that it seems very unlikely that BQP includes all of NP. However, some problems which are not known to be in P have been shown to be in BQP. The most notable of these is factorization, which is the fundamental concept behind the security of RSA.

Having discussed the distinction between these three important complexity classes (P, BQP, NP), the course can then explore at a high level what quantum algorithms have been proposed and what current security protocols stand at risk if these quantum algorithms are realized. The most popular quantum algorithms that should be covered are Shor’s Algorithm and Grover’s Algorithm (NAP, 2019). This discussion can also help demonstrate which security algorithms are not likely to be broken by quantum computing. Hash algorithms such as SHA256 are significantly less susceptible to being broken by quantum than encryption algorithms such as RSA and ECDSA.

At the conclusion of the course’s discussion of quantum computing it will also be helpful to briefly cover the benefits that quantum computing could bring to cryptography. For example, generating secure random numbers is fundamental to almost any cryptographic algorithm. Quantum computers offer the potential for the first time to generate truly random numbers (Lipman, 2021).

In the final day or two of class, I suggest covering AI as another major emerging technology that will have an impact on cybersecurity. Cybersecurity can be improved dramatically by AI that is capable of recognizing patterns of malware and identifying potential solutions (Bresnicker et al., 2019). Currently, one of the most common practices of anti-malware is using a library of software signatures of known malware. However, this strategy is ineffective against new malicious programs that are not in the database. Muppidi, the Chief Technology Officer at IBM Security, has also explained how AI can be used to assess any given user's risk and adjust security measures accordingly (IBM, 2022). This can be incredibly helpful to develop more secure protocols that are less likely to get in the way of users' experience, and thus more likely to be implemented.

4. Expected Benefits

CS 4630 is the most in-depth cybersecurity class taught at the undergraduate level at UVA. As such it is the best training offered by UVA to any CS majors planning to enter the cybersecurity workforce. Given that it is a one semester course and cybersecurity and malware are quite complex topics, the course as it is currently written necessarily focuses on simpler topics, using examples from an older era of malware. However, given the speed at which cybersecurity evolves, providing students a taste of where the field is headed will give them a chance to see how the principles taught in this class are applicable in the present day and better prepare students for future jobs within cybersecurity.

5. Conclusion

There can often be a temptation to view cybersecurity as an isolated field within Computer Science. Particularly a higher level class like Defense Against the Dark Arts can feel very focused on ultra-technical concepts such as assembly code or memory storage. In addition to preparing students for future evolution in the field of cybersecurity, these proposed changes will demonstrate how the field is interconnected with other concepts studied in Computer Science. The brief analysis of AI shows how other branches of CS can impact cybersecurity, and a discussion of quantum computers and runtime analysis can shed light on how even more abstract concepts from Algorithms can find practical application in this field that is commonly regarded as more technical.

6. Future Work

This project is designed to give a high-level proposal of how CS 4630 could be changed to include an analysis of how rising technology will affect the field. In order to implement these changes, more detailed study would be required to determine which topics are most beneficial to include in the course. Most likely an exploration of how to break RSA with a quantum algorithm along with a demonstration of how AI can be used to catch malware would be among the most worthwhile topics to cover. However, there is a broad range of subjects relevant to this purpose which should be explored in further detail.

References

Aaronson, S. (2010) BQP and the Polynomial Hierarchy. *Association for Computing Machinery, p. 141–150.* https://doi.org/10.1145/1806689.1806711

Aaronson, S. (2021) What Makes Quantum Computing so Hard to Explain? *Quanta Magazine* https://www.quantamagazine.org/why-is-quantum-computing-so-hard-to-explain-20210608/

Bitcoin (2021). Elliptic Curve Digital Signature Algorithm. https://en.bitcoin.it/wiki/Elliptic_Curve_Digital_Signature_Algorithm

Bresnicker et al. (2019). Grand Challenge: Applying Artificial Intelligence and Machine Learning to Cybersecurity. *Institute of Electrical and Electronic Engineers.* https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8909930

Brett et al. (2017). Artificial Intelligence for Cybersecurity: Technological and Ethical Implications. *Center for Cyber and Homeland Security at Auburn University.* http://www.jstor.com/stable/resrep21461

Google (2022). Email Encryption FAQs. https://support.google.com/transparencyreport/answer/7381230

Lipman, P. (2021). How Quantum Computing Will Transform Cybersecurity. *Forbes Magazine.* https://www.forbes.com/sites/forbestechcouncil/2021/01/04/how-quantum-computing-will-transform-cybersecurity

Mermin, N. D. (2007). Quantum Computer Science: An Introduction. *Cambridge University Press.* https://doi.org/10.1017/CBO9780511813870

National Academies Press (2019). Quantum Computing's Implications for Cryptography. https://www.nap.edu/read/25196/chapter/6

Singh, S. (1999). The Code Book. *Doubleday.*

Wilcox, J. (2006). Solving the Enigma: History of the Cryptanalytic Bombe. *Center for Cryptologic History, National Security Agency.*