Proposing a New Course: Modern Computing and Security Practices

A Technical Report submitted to the Department of Computer Science

Presented to the Faculty of the School of Engineering and Applied Science

University of Virginia • Charlottesville, Virginia

In Partial Fulfillment of the Requirements for the Degree

Bachelor of Science, School of Engineering

Caroline Ehler

Spring, 2021.

Technical Project Team Members

Daniel Keith

On my honor as a University Student, I have neither given nor received unauthorized aid on this assignment as defined by the Honor Guidelines for Thesis-Related Assignments

Aaron Bloomfield, Department of Computer Science

Proposed CS Course
CS4501: Modern Computing and Security Practices

Caroline Ehler
Computer Science
University of Virginia
Charlottesville, VA United States
cee7zm@virginia.edu

Daniel Keith
Computer Science
University of Virginia
Charlottesville, VA United States
dmk3pc@virginia.ed

ABSTRACT

The new computer science (CS) course we propose will cover the ethics of security and provide a high-level overview of computer security. We will include modern practices that will pertain to as many students as possible, instead of simply looking back at security practices that students already know and study in other required courses. In addition, an important part of the class will consist of open discussions about privacy rights, opinions on personal security, and related topics. Many UVa CS students take classes such as Computer Architecture, Advanced Software Development, and various STS courses, but they may not have learned how to connect the courses pertaining to ethics with those pertaining to cybersecurity. This course will synthesize these two topics of study for a more holistic and complete cybersecurity education. As UVa CS students will enter many different professional fields or areas of higher research, every graduate should have a fundamental understanding of security, privacy, and ethics, regardless of their career path. We believe there is currently a gap in the curriculum, as students' STS classes cannot solely focus on CS topics, and this class would help to fill that gap.

The course will cover two weeks' worth of material including lectures, homework assignments, and readings. The material will provide a short introduction to the course, but the bulk of the material will cover the topics previously stated in-depth. We hope that the material and educational style of this course will be both engaging and informative for students; if the students cannot meaningfully engage with the topics and with each other, then the course did not fulfill our objective. It is our desire that students who complete this course will walk away with both a better understanding of the technical world around them and a mindset that will protect them and their future colleagues as they pursue careers in the field of computing and beyond.

INTRODUCTION

The Computer Science curriculum at UVa is certainly rigorous, and the combination of required core classes plus a handful of electives that students can choose from ensures that students receive a well-rounded and personalized education. However, we feel that there is currently a gap in the curriculum that disadvantages students when they eventually graduate and enter the computer science workspace. In today's society, technological failures and security breaches constantly make up front-page news stories; building secure, reliable hardware and software is more important than ever. As it stands, there is not a CS course offered at UVa centered around modern security practices and the ethics of secure programming. These topics are discussed sporadically across several classes, as mentioned in later sections, but the course we propose, solely focused on practical security issues, would equip UVa graduates for situations that they are likely to encounter in their careers.

BACKGROUND

Before introducing the core of our proposed CS course, it is important to note what courses UVa students may take prior to this one. Computer science students in the engineering school are required to take two introductory coding courses and a program and data representation course, CS 2150, before taking CS electives. Security is more heavily introduced in the core classes at the 3000 and above levels such as the required courses of Computer Architecture, Advanced Software Development Techniques, and Operating Systems. These courses touch on security briefly. The security topics covered are not in-depth and are very specific to the security issues related to that course. Additionally, the BSCS students are required to take five electives in addition to their required core courses.

RELATED WORK

UVa's Department of Computer Science offers many electives for students to fill their five required elective courses. Many of the courses offered teach more in-depth about security than the core courses do, but the topics are still very specific to the course. For example, both the Cloud Computing and Databases classes integrate security into the curriculum and explain a variety of security practices

and vulnerabilities that are relevant, but only share security tactics about cloud security and database security.

CS 3710 Introduction to Cybersecurity offers an overview of many different vulnerabilities found in virtual systems, cyber-attacks, how to conduct and prevent them. This course's curriculum covers well-known cyber-attacks and good security practices, which are important to know, but are somewhat out-of-date. The curriculum is not obsolete; it is important to know how to prevent attacks like SQL injections or binary exploitations. However, the UVa CS electives do not seem to cover more modern security topics a student might be looking to learn more about.

CS 4753 eCommerce, a course no longer offered, taught about web security and more modern topics and technologies such as biometrics, cryptocurrency, NFC, data encryption, password cracking, etc. Our proposed CS course is loosely based off of this course's approach to teaching modern technologies and proper security practices.

In addition to security, our course also focuses on ethics. BSCS students are required to take a course called STS 4600 which teaches ethics to fourth year students before graduating and heading to the workforce where they may face ethical dilemmas. While taking this course, we believe it beneficial to introduce ethics into computer science discussion earlier than the final STS course. Very little CS-specific ethics are discussed in the currently-offered courses, which is a missed opportunity to integrate ethics and CS. STS 4600 is offered to all engineering students, so CS students do not have a chance to discuss solely CS-specific topics or technologies. The STS course focuses more on ethics with an application of technology, rather than integrating computer science with ethics.

These courses don't cover the entirety of the parts of modern security practices we wish we could have learned. Making a new CS course that introduces these important and very relevant security topics in conjunction with ethical discussions prepares students for the workforce well. Security and ethics can often go hand-in-hand, and students' learning can benefit when the topics are combined.

SYSTEM DESIGN

1 Overview

Proposing a new computer science course about security, we identified new topics to be covered in the course that focus on modern security applications such as cloud computing, mobile development, cryptocurrency, and E-commerce. Our security course's topics are tied together with ethical analysis of each topic and security practices to develop students' ability to assess ethical security decisions, preparing them for their future endeavors in technology.

The course is designed for a Tuesday/Thursday class where lectures are estimated to take one hour and 15 minutes. The course is to cover the following topics that are divided into about two-week units: Overview of Cybersecurity and Ethics, Cloud & Mobile Security, Modern Security Issues & Techniques, Cryptocurrency, Blockchain & E-commerce, and Security & the Future. The rest of the semester's classes will be allotted for students to present on a final presentation during class time.

The course is designed with lectures, weekly quizzes, homework assignments (about one per unit), a midterm and final exam, and a final presentation. All of the key topics will be synthesized by students in a final presentation where they will choose a real-world example of a security breach or malfunction and analyze the strengths and weaknesses of the response, as well as discussing the ethics of the situation using frameworks and concepts presented throughout the semester.

The capstone project deliverables only include a syllabus, four lectures, four readings, two homeworks, and two quizzes. Should this course be implemented in the UVa computer science curriculum, the syllabus provides both a course overview and breakdown. The sole prerequisite is CS 2150 so as to verify students have the abilities to complete all coding homework assignments. Though it is not required, we suggest students take CS 3710, CS 4630, and or CS 4740 to round out their knowledge of security as these courses will pair well with our proposed course. The four lectures we have created come from different units so that a variety of information can be provided to give a better synopsis of the class through the capstone project. All course material can also be found and reviewed through a Collab page which was setup to imitate a real Collab course. The Collab page is where students would take weekly quizzes, find homework instructions and turn in assignments.

The following sections cover each of the deliverables presented in this capstone project. A synopsis of the lectures, homeworks, readings and quizzes are presented, explaining the design decisions for the course.

1.1 Lecture I - Ethics

The first unit of our course serves as an introduction to the general concepts that will be discussed throughout the semester. These concepts include cybersecurity, cloud computing (the platform for many modern security and privacy concerns), and

the topic of our first capstone lecture, various forms of ethics in relation to the world of computing.

The lecture first introduces a classic ethical dilemma, "The Trolly." In this scenario, a bystander is forced to decide between letting a runway train kill several people tied down to a railroad or pushing a large man into the train's path, saving the people tied to the tracks but killing the man in the process. Students do not need to have a clear and concise answer to this question, but the activity serves to introduce the type of thinking that will be developed throughout the lecture and the course.

Different ethical frameworks make up the bulk of the lecture material. The focus of this lecture is on several normative frameworks: Virtue Ethics, Consequentialism, and Deontology. Virtue ethics is centered around the idea of virtues, or favorable traits of characters; moral virtues are what should drive all decisions. It was pioneered by Greek philosophers such as Socrates and Plato, and over the centuries, scholars have defined many ways to tell "right from wrong."

Consequentialism holds that the most important factor in decision making is the resulting consequences of each alternative. It is often confused with utilitarianism, which is the idea that actions that increase the overall net good of society are actions worth taking. Consequentialism is simply one aspect of utilitarianism, as it narrows a similar concept down to the scope of each individual act. On its own, consequentialism is an inherently vague framework; it relies on how one chooses to define a "good consequence." The simplest definition of "good" is the hedonistic view, where pleasure is good and pain is bad. There are many more ways to define "good," two of the most common being pluralism, where knowledge is good and deception is bad, and the utilitarianism of rights, where an act is good if it respects a certain moral right.

Deontology is based around moral norms and the concept of duty. It states that all people have a duty to uphold certain moral standards. Deontology serves as a contrast to consequentialism; acts can be defended if there is a moral justification, even if the consequences of the act are not universally considered good. It also leaves room for agents to take action that favors people or things that matter to them; many other ethical frameworks do not prescribe that way of thinking. The two main forms of deontology are based on perspective. Agent-centered deontology holds that agents have a duty to take action or refrain from an action, and patient-centered deontology holds that we all have a duty to respect the will and choices of others, as they hold the same rights and duties that we do.

Based on these three frameworks, the class then divides into three groups to build different analyses of example ethical dilemmas related to computing. These scenarios range from building privacy features to encountering morally compromised clients. This lecture serves as an introduction to how students will approach the content of the course and gives practical examples that they could run into in their future career paths.

1.2 Lecture II - DevSecOps

As a part of the cloud and mobile security unit, the second capstone lecture covers DevSecOps. This topic is important for computer science students to know before heading into the workforce as many companies have implemented this practice into their development processes and thought processes. Students benefit from learning about DevSecOps, developing and strengthening their mindset regarding security so as they code at internships or their time at UVa, they practice safe security measures and do not separate security from development.

The DevSecOps lecture begins with an overview of how the COVID-19 pandemic has affected security, emphasizing the importance of security and creating a sense of urgency to this relevant problem in technology. It is noted that certain cyberattacks have risen in the pandemic relating to the increase of individuals online and organizations working from home. Multiple types of security attacks are covered to introduce some pandemic-related attacks. Students are then asked to complete an in-class activity to perform research on the topic of COVID-19 cybersecurity to find and report on an article relevant to what excites them. This activity will further reinforce the importance of security when applied to an industry, company, or any given example the student deems interesting.

The concept of DevOps is introduced to explain how products, companies, and consumers benefit when development teams are integrated with IT operations teams when developing a product. This process and mindset can prevent vulnerabilities, but still lacks experts in security. This leads to the introduction of DevSecOps and presents the case for security where DevOps lacks. DevSecOps practices include integrating development, IT operations, and security teams when creating and deploying technical products to build-in proper security practices. DevSecOps should be a priority since the added burden of the pandemic can lead to poor execution on development teams that may already be overwhelmed, in which the other integrated teams can help protect against vulnerabilities and act as a check. An IBM video may be played during the lecture introducing the topic as an overview to prime students before diving deeper into DevSecOps, should time allow. The idea of continuous integration and continuous deployment

is introduced to show how DevSecOps is a continuous loop or cycle. The benefits to DevSecOps are covered as well, including team awareness, improved collaboration across teams, lack of silos, and saving time and money.

Lastly, the lecture covers what will be offered as a reading before class on relevant DevSecOps examples in companies they may be familiar with or may be interested in to again, reinforce the importance of this topic and to best prepare them for their future careers in technology.

1.3 Lecture III - Modern Viruses and Vulnerabilities

As part of the Modern Security Issues & Techniques unit, our third capstone lecture covers a wide range of different data breaches, viruses, and intentional attacks on computer vulnerabilities. After each section of the lecture, the class will have a smaller discussion around a guiding question related to the previously mentioned material. This lecture puts an emphasis on modern vulnerabilities rather than more classically taught vulnerabilities not only because that is the greater scope of the course, but also because anecdotes that students can relate to and understand the impact of will have a greater effect on how they view security practices in the future.

The lecture is divided into three categories, labeled as Defcon 3-1, based on the severity of the attack. The first category covers personal data breaches, highlighting the attacks on iCloud and Facebook. In 2014, hackers were able to brute-force their way into many celebrities' iCloud accounts by simply guessing passwords until they logged in, as the login system did not have a limit on login attempts per account. They also sent phishing emails pretending to be iCloud workers to convince people to send in their security details. Over 100 individuals had personal data released to the public, with private nude photographs of popular female celebrities as the main target content. Facebook suffered a similar attack in 2018, where the personal information of 50 million users was exposed, including Mark Zuckerberg himself. The attack served as another call to action for companies to ensure that their software designs were safe before pushing them out to the public; as the commissioner of the FTC, Rohit Chopra, said, "The cost of inaction is growing, and we need answers."

The second category covers the types of attacks that can cause serious, tangible damage, whether that be to an infrastructure or to a person. In 2014, Cesar Cerrudo, a professional white-hat hacker, discovered vulnerabilities in popular traffic control systems. The system in question, the Sensys Networks VDS240, communicated all signals in plaintext with no encryption, and the wireless transmissions were not adequately protected. Because of this, Cerrudo could break into a traffic intersection's sensor from up to 1500 ft away and send his own signals, which could lead to gridlock if lights were kept red or collisions if lights were kept green. Another white-hat hacker named Barnaby Jack conducted similar research focused on the insulin pump and the pacemaker. Jack reverse engineered both the insulin pump and the pacemaker and found methods of infiltrating both remotely. For the insulin pump, he could stop the flow of insulin or deposit an unhealthy level of insulin at once, both of which carry potentially fatal consequences. He could perform this attack from up to 300 ft away. For the pacemaker, Jack was able to take control from up to 50 ft away and could shut off the device or deliver a sudden high voltage shock that would instantly kill the victim. These technologies may be revolutionary, but without adequate security systems, medical devices could cause more harm than healing.

The final section of the lecture involves several instances of "e-Warfare," scenarios where cyberattacks were deliberate acts of war or terror. One of the oldest occurred in 1982 in the heat of the Cold War, known as "the original logic bomb." The US hid a "Trojan Virus" inside a software package they suspected the USSR would steal, designed to cause small failures along the Russian pipeline. However, the bomb acted more violently than expected, and actually caused the largest ever non-nuclear explosion viewable from outer space! In 2003, technical failures in the US energy grid caused a blackout across much of the Northeast of America, including sections of Canada. These failures included inadequate electrical systems, a lack of scaling to match growth of the customer base, and a critical race condition in GE's energy management system that was not addressed until it was too late. In 2008, the US military was hacked thanks to a suspicious USB drive found in the parking lot of an international military base. When the drive was plugged into a computer connected to the central Department of Defense database, the worm known as agent.btz spread, siphoning data and opening new backdoors for attack. The lecture's last example of e-Warfare is Stuxnet, a worm that was designed to target programmable logic controllers. It was designed in such a fashion that it would be mostly invisible to antivirus security, as it lay dormant until it was instructed to attack. The most famous application of Stuxnet was used to destroy thousands of Iranian nuclear centrifuges in 2010, setting their nuclear program back by several years. Forms of Stuxnet are still in existence, and while much more has been discovered about its nature, hackers are still able to deploy it effectively.

1.4 Lecture IV - Cryptocurrency

The fourth lecture covers cryptocurrency, coming from the unit about Cryptocurrency, Blockchain & E-commerce. This lecture is the introductory lecture on cryptocurrency, the blockchain, and NFT's.

The lecture begins by explaining Aristotle's Sound Money to build a foundation for understanding cryptocurrency. For currency to be considered "Sound Money," it should possess seven attributes Aristotle compiled: durability, transferability, divisibility, intrinsic value, scarcity, recognizability, and fungibility. The currency should be physically stable (durable), easy to transfer to another person (transferable), subdivided into smaller units (divisible), have a limited quantity available (scarce), easy to recognize and verify authenticity (recognizable), and may be substitutable for another unit of money (fungible). These attributes are held by cryptocurrency in unique or abstract ways. The lecture is to present how cryptocurrency possess these attributes, provide examples, inform students on relevant technologies, and explain how vulnerable or secure these currencies are.

To explain how cryptocurrency uniquely possesses these attributes, virtual scarcity is introduced. Virtual scarcity is enforced not by banks or physical resources like more tangible currencies, but rather based on a hash function. Here, Bitcoin is introduced, a currency where scarcity is based on a hash function. Bitcoin is a distributed, decentralized digital currency system based on a hash function and consists of a chain of digital signatures. A short video is played to drive home the introductory points of Bitcoin.

As Bitcoin relies on scarcity, an additional way to do so is to provide a reward for those that discover new hashes to new bitcoin, also called "mining" for Bitcoin. Hash functions are irreversible, and miners can compute hashes for ledger blocks that fall below a special value called the difficulty target. A miner may try various "nonces" until one computes a valid signature, in which one has successfully mined Bitcoin. To create artificial scarcity, when a new nonce is found, the difficulty target is adjusted to keep Bitcoin from being mined too quickly. Nonces are further explained through nonce and hash examples.

To introduce an important and unique technology of cryptocurrencies, the blockchain is introduced, emphasizing the security aspects and tying it back to Bitcoin. Bitcoin transactions are recorded in blocks which contain the signature of the previous block, linking them together in a chain. This allows for only pseudonymity and retroactive data mining. Bitcoin can be authenticated with public keys, hold integrity with digital signatures and a cryptographic hash, and allow confidentiality through the blockchain's pseudonymity.

Next, Non-Fungible Tokens, or NFTs, are introduced to compare with Bitcoin. NFTs are another form of cryptocurrency and possess Aristotle's attributes of Sound Money. They are unique, provably authentic, scarce tokens utilizing blockchain technology. They are on the rise currently and differ from Bitcoin as an NFT is non-fungible and cannot be interchanged with another. This means each NFT holds a different value from another. The unique aspect to NFTs that allows for this is that each is an expression of virtual art such as a digital image, GIF, video, etc.

To close, the class is given time to discuss the presented ethics of cryptocurrency. Issues such as roadblocks prevent mass adoption like inaccessibility, volatility of transaction fees, etc., distrust in banking, and cryptocurrency's carbon footprint. Miners use so much computational power that it can increase electricity consumption. Miners head to where power is cheap. These ethical concerns are important to consider when purchasing, endorsing, or developing for these technologies.

1.5 Homeworks

For the course, we created two homework assignments, one for each week of lecture material created. The first assignment is based off of the Modern Viruses and Vulnerabilities lecture and involves researching the Heartbleed virus, a recently famous vulnerability in the OpenSSL cryptographic software library. Students are tasked with doing research on Heartbleed using attached readings and their own sources, using their research to answer several short answer questions, and then complete a tutorial-style lab with a brief writeup to submit. By the end of this assignment, students will be able to both explain the fundamental properties of the Heartbleed vulnerability and demonstrate how to exploit the vulnerability in a safe, isolated environment.

The second homework assignment is in conjunction with the cryptocurrency unit. The cryptocurrency lecture covers NFTs, but the bulk is about Bitcoin and the blockchain. To supplement for NFTs, the homework assignment is to follow the given tutorial to create an NFT. The tutorial doesn't require real currency but rather provides a fake wallet of money. Students are expected to turn in a PDF document with the requested information of them, showing they completed the tutorial. Students will receive hands-on experience with cryptocurrency through a coding tutorial and gain extremely relevant and modern experience with NFTs.

1.6 Readings

Students are highly encouraged to read the readings given out before each lecture. Quiz material may contain questions about the readings, providing incentive students to read them. The four readings we have prepared correspond to the four lectures we created. The reading for Lecture I consists of articles from the Stanford Encyclopedia of Philosophy covering the three ethics frameworks in much greater detail than can be accomplished in class. For the second lecture on DevSecOps, a reading is given about real-world examples of how large-scale tech companies that deployed DevSecOps teams were successful in their goals. The third lecture's reading is in tandem with the homework assignment—research on the Heartbleed vulnerability. The reading for Lecture IV on cryptocurrency is the Wikipedia page on NFTs, providing an overview of the technology for students to prime them for the lecture and the homework assignment they must complete on NFTs.

1.7 Quizzes

Students are expected to complete and submit weekly quizzes about the current week's lecture material. The quizzes are found on the Collab site, where students will take them and turn in. All quizzes are open-note, so students may look back to the week's lectures, but not all of the material stated in the lectures are stated on the slides, requiring students to still pay attention in class, encouraging them to take notes. The two quizzes we have created are on ethics and cryptocurrency. All quizzes contain around five questions, and are typically multiple choice or short-answer.

RESULTS

Looking back at our course outline and materials, we feel that we successfully accomplished our objective to create a class focused on modern computing practices with an emphasis on security and ethics. We showed the lectures and syllabus to some of our peers (one majoring in CS and another not majoring in CS) and they both found the material to be engaging and worthwhile as a course. A fellow CS student commented after reviewing the course material that the course we proposed achieves the goal to bridge the security and ethics gap. She believes that the current classes offered teach either introductory concepts or give basics on how to perform attacks. She concluded that most current security teachings focus more on specific weaknesses to avoid, rather than a comprehensive methodology for creating secure systems as a whole. As mentioned in Lecture III, the best way to combat security breaches is to stop them before they can even attack!

Even though the non-CS student was mostly unfamiliar with the CS curriculum, he could see how the course filled the gap that we have observed. Of course, in this context, the only real way to know if the course was successful would be to actually implement it as a full class with enrolled students, more lectures and homeworks, etc. Nevertheless, based on the limited scope of the project, we believe our proposed course is a worthwhile addition to the CS curriculum at UVa.

CONCLUSIONS

We designed a system of course material to meet the need of the gap in the UVa CS curriculum. The course bridges the gap in curriculum between ethics and modern security practices. The material we have created provides a synopsis of the course through two weeks' worth of course material and a syllabus overview of the course. The course's goal is to be both engaging and informative for students regarding relevant topics in security and support them through practical ethical experience. It is our desire that students who complete this course will walk away with both a better understanding of the technical world around them and a mindset that will protect them and their future colleagues as they pursue careers in the field of computing and beyond.

FUTURE WORK

If we had the time, we would finish the course curriculum and materials needed to teach the course. We would also have created, tested, modified, and customized the homework assignments to our liking. With more time, we would like to survey students to gauge interest in a course like this and modify it before offering the course. Once all of the materials are finished, we would offer the course to CS students as an elective course.

REFERENCES

[1] Lau, K. (2020). Non Fungible Tokens. Crypto.com. https://assets.ctfassets.net/Crypto.com_Macro_Report_-_Non-Fungible_Tokens.pdf

[2] Boscovic, D. (2021, March 31). How nonfungible tokens work and where they get their value – a cryptocurrency expert explains NFTs. The Conversation. https://theconversation.com/how-nonfungible-tokens-work-and-where-they-get-their-value-a-cryptocurrency-expert-explains-nfts

[3] Gilfoyle, B. (2020). Cryptocurrency. http://www.piedpiper.com/app/themes/pied-piper/dist/images/Cryptocurrency_Presentation.pdf

[4] Poremba, S. (2020, May 11). Businesses Underestimate COVID-19 Cybersecurity Risks. Security Boulevard. https://securityboulevard.com/2020/05/businesses-underestimate-covid-19-cybersecurity-risks/

[5] Goldstein, D. (2021). How Cybercriminals Take Advantage Of COVID-19 | BrandShelterTM. BrandShelter. https://www.brandshelter.com/en/news/how-cybercriminals-take-advantage-of-covid-19

[6] IBM. (2020). The COVID-19 cyberwar: How to protect your business. https://www.ibm.com/thought-leadership/institute-business-value/report/covid-19-cyberwar

[7] Reblaze. (n.d.). What is DevSecOps? Reblaze.Com. https://www.reblaze.com/wiki/devops/what-is-devsecops/

[8] Charlton, A. (2015, January 2). iCloud accounts at risk of brute force attack as hacker exploits "painfully obvious" password flaw. International Business Times UK. https://www.ibtimes.co.uk/icloud-accounts-risk-brute-force-attack-hacker-exploits-painfully-obvious-password-flaw-1481623

[9] Zetter, K. (2017, June 3). Hackers Can Mess With Traffic Lights to Jam Roads and Reroute Cars. Wired. https://www.wired.com/2014/04/traffic-lights-hacking/

[10] Alexander, W. (2013, June 25). Barnaby Jack Could Hack Your Pacemaker and Make Your Heart Explode. Vice. https://www.vice.com/en/article/avnx5j/i-worked-out-how-to-remotely-weaponise-a-pacemaker

[11] Loney, M. (2004, March 1). US software "blew up Russian gas pipeline." ZDNet. https://www.zdnet.com/article/us-software-blew-up-russian-gas-pipeline/

[12] Wikimedia Foundation. (2021a, February 2). Agent.BTZ. Wikipedia. https://en.wikipedia.org/wiki/Agent.BTZ

[13] Wikimedia Foundation. (2021b, February 28). Stuxnet. Wikipedia. https://en.wikipedia.org/wiki/Stuxnet

[14] Zetter, K. (2017a, June 3). An Unprecedented Look at Stuxnet, the World's First Digital Weapon. Wired. https://www.wired.com/2014/11/countdown-to-zero-day-stuxnet/

[15] Isaac, M., & Frenkel, S. (2019, March 19). Facebook Security Breach Exposes Accounts of 50 Million Users. The New York Times. https://www.nytimes.com/2018/09/28/technology/facebook-hack-data-breach.html

[16] Virtue Ethics (Stanford Encyclopedia of Philosophy). (2016, December 8). Stanford Encyclopedia of Philosophy. https://plato.stanford.edu/entries/ethics-virtue/

[17] Consequentialism (Stanford Encyclopedia of Philosophy). (2019, June 3). Stanford Encyclopedia of Philosophy. https://plato.stanford.edu/entries/consequentialism/

[18] Deontological Ethics (Stanford Encyclopedia of Philosophy). (2020, October 30). Stanford Encyclopedia of Philosophy. https://plato.stanford.edu/entries/ethics-deontological/