Oct 12th, 10:30 AM - 10:55 AM

Adversarial Thinking: Teaching Students to Think Like a Hacker

Frank Katz
Georgia Southern University, fkatz@georgiasouthern.edu

Abstract
Today's college and university cybersecurity programs often contain multiple laboratory activities on various different hardware and software-based cybersecurity tools. These include preventive tools such as firewalls, virtual private networks, and intrusion detection systems. Some of these are tools used in attacking a network, such as packet sniffers and learning how to craft cross-site scripting attacks or man-in-the-middle attacks. All of these are important in learning cybersecurity. However, there is another important component of cybersecurity education – teaching students how to protect a system or network from attackers by learning their motivations, and how they think, developing the students' "abilities to anticipate the strategic actions of cyber adversaries, including where, when, and how they might attack, and their tactics for evading detection."

This paper describes the content and implementation of a 6 hour 15 minute (5 class sessions) module in Adversarial Thinking in a Network Security course, the students' perceptions of the value and importance of the module as a result of their anonymous responses to a survey on the module, and the statistical results of a Data Breach Pretest-Posttest Assessment to measure how well they understood the concepts involved in Adversarial Thinking as part of learning cybersecurity.

Location
KSU Center Rm 460

Disciplines
Curriculum and Instruction | Information Security | Technology and Innovation

Comments
Key words: Cybersecurity education, Adversarial thinking, Game theory, Behavioral game theory, Dominant strategies, Utility preferences, Interdependent choices

INTRODUCTION

At the annual symposium of NSA-CAE institutions in Miami, Florida, in November 2018 I was introduced to the Clark website, which contains many different modules in Cybersecurity education, all created by faculty at the various NSA-CAE institutions. The depth of these modules range from just a lecture or two on a topic, to several weeks' worth of material, to entire courses. The available topics include everything from lectures on social engineering to labs. The site was created and is maintained by NSA-CAE institution Towson University.

Although my university has successfully used the material from the various publishers of our texts, we are always looking for new and original material to supplement our existing curriculum in cybersecurity. Material from the Clark site seemed to be quite suitable for this purpose, varying from entire courses, to one to three week modules, to just one to two hour lessons. Included in the Clark site are many different cybersecurity lab exercises, but one particular short module stood out as what could be a valuable addition to our 4000 level Network Security or Ethical Hacking classes – a module in Adversarial Thinking, or how to think like a hacker.

The materials in the adversarial thinking module, including PowerPoint slides, exercises, and instructor notes and scoring examples, were obtained from the Clark site, and were created by Professor Seth Hamman at Cedarville University, in Cedarville, Ohio. Together with Professor Ken Hopkinson, the description of the curriculum and findings from their work, which is referenced throughout this paper, was published in the Journal of the Colloquium for Information System Security Education (CISSE) in September 2016.

This paper will explain what adversarial thinking is, the concepts underlying it and some of the exercises included in the module to teach the model. It will describe the implementation of the module in a Network Security class during the spring, 2019 semester at Georgia Southern University, the statistical results of an exercise in that module, and the results of a survey of the students regarding the module. Based on that data, a conclusion will be drawn as to the effectiveness of the module in teaching Adversarial Thinking.

Why Teach Adversarial Thinking?

Starting with our Fundamentals of Information Systems Security course, and throughout our cybersecurity curriculum, there are various references to the "hacker mindset". Hackers are categorized, based on their intent as "Black-hat", "White-hat", or "Gray-hat" hackers. (Oryiano and Solomon, 2008, p. 6-7) Hacker motivation, whether it is financial, to demonstrate the ability to prove that the hacker is more powerful than defenders, or whether the activity is just because it is an exciting challenge, is taught in our Network Security and Ethical Hacking courses. So it is natural that we attempt to teach students how to out-think their opponent. That ability is known as adversarial thinking.

Indeed, as A. McGettick (2013) stated (as cited in Hamman and Hopkinson, 2016, p. 2) , "a team of subject matter experts convened by the Association of Computing Machinery (ACM) to identify cybersecurity curricular guidelines agrees that teaching adversarial thinking is vital." The summary report of those subject matter states, "To protect systems…we need to temporarily adopt the thinking process of the malevolent hacker…Developing this way of thinking must be part of [emphasis added]…educating cybersecurity professionals."

In his paper, Fred Schneider (2013) wrote (as cited in Hamman and Hopkinson, 2016, p. 2), adversarial thinking is "the very essence of game theory. In it, actions by each player are completely specified; for cybersecurity and safety-critical systems, identifying possible player actions is part of the central challenge."

Although not a requirement when our university applied for NSA-CAE designation in 2015, the NSA-CAE required Knowledge Units (KUs) now includes in its Non-Technical Core – Cyber Threats (CTH) KU, the requirement that students should be able to "identify the bad actors in cyberspace and compare and contrast their resources, capabilities/techniques, motivations and aversion to risk." (NSA CAE-CD 2019 Knowledge Units, 2019) Consequently, it is not only important to teach adversarial thinking so that students can learn to out-think hackers and stay one step ahead of them, teaching it is a requirement for any college or university attempting to become designated or re-designated as an NSA-CAE/CDE.

CONTENT OF THE ENTIRE MODULE

The module consists of three primary lessons: an introduction to adversarial thinking; an introduction to game theory; and an introduction to behavioral game theory. In the notes to the instructor, it is stated that each lesson can be taught in one hour, or three hours total. In reality, with the time it took to describe, do, and discuss the results of each exercise in their entire module of three lessons, it took five class sessions of 75 minutes each, or six hours and fifteen minutes to complete. In this section, each lesson is briefly described, as well as some of the student exercises used to teach the concepts.

Lesson 1 – Introduction to Adversarial Thinking

This lesson defines many of the terms that will be used in the module, its learning outcome being that after this lesson, "students will be able to analyze cybersecurity from the strategic perspective of cyber adversaries."

It began by having the students perform the Data Breach Exercise, used as a pretest. The Data Breach Exercise will be described in greater detail below, but it is an important exercise because it is used a posttest at the end of the module to quantify how well the students learned the concepts.

This lesson narrowed the definition of a computer hacker, especially with regard to the commonly taught "C-I-A Triangle" of confidentiality, integrity, and availability of systems and data. It introduced the "3 B's of Security," components which are contained in "every security context:

- Bounty – valuables that must be protected from bad guys
- Bad Guys – persons who want to get their hands on the bounty, and
- Barriers – obstacles placed between the bad guys and the bounty."

Lesson 1 made the valid point that without an adversary, an opponent, there is no need for cybersecurity. Given that all systems contain "bounty," there are bound to be adversaries, or "bad guys," and thus the definition of adversarial thinking is "the ability to think like a hacker." (Hamman, 2018, Lesson 1 slides, slide 13) In slides 14 through 16, the lesson then proceeds to use cognitive psychology, the "study of higher mental processes, such as attention, language use, memory and perception, problem solving, and thinking to more precisely define what it means to think like a hacker. In using Sternberg's Triarchic Theory of cognitive intelligence, that "there are

three distinct aspects of the intellect, analytical, creative, and practical" they further refine the definition by discussing how much book smarts (analytical), creative thinking (the ability to "make new and unique connections"), and practical intelligence (the ability to "plan, strategize, and accomplish goals") affect the activities of a hacker. (Hamman, 2018, Lesson 1 slides, slides 14-16) Consequently, their more precise definition is that "adversarial thinking is the ability to embody the technological capabilities, the unconventional perspectives, and the strategic reasoning of hackers." (Hamman, 2018, Lesson 1 slides, slide 18)

Lesson 2 – Introduction to Game Theory

The second lesson introduced students to the concept of game theory, its learning outcome being that "students will be able to analyze a strategic scenario from a game theoretical perspective." While game theory has been prominently used in economics, it can be applied to many different disciplines, from cybersecurity to warfare. According to David Levine of UCLA, there are two types of game theory, cooperative, and non-cooperative. Naturally we are discussing non-cooperative game theory, which "deals with how intelligent individuals interact with one another in an effort to achieve their own goals." (Levine, n.d.)

Several different exercises were employed in this lesson to teach the students how game theory could be used to prevent an adversary from hacking a system. In each exercise, the students were divided into groups, and the groups were required to evaluate the scenario from a game theory perspective and report their conclusions.

Before performing the exercises, students were given the rules and several definitions. The first definition is of *game theory*, "a mathematically rigorous approach to analyzing strategic contests (not games of skill or chance)." For our exercise, it can be defined as "the study of *interdependent* decision making between multiple *players* where each player strives to maximize his *utility.*" (Hamman, 2018, Lesson 2 slides, slide 7) Our three subsequent definitions are:

- *Players* – the actors in the game.
- *Interdependent choices* – the final outcomes for each player are dependent on all of the other player's choices.
- *Utility preferences* – an ordering of the outcomes for each player from least desirable to most desirable.

The exercise described below was an actual historical scenario from World War II. Entitled "The Battle of Bismarck Sea," in which the Japanese fleet could sail a major convoy north of the island of New Britain, where they would encounter rain and poor visibility, or they could travel south of the island, where the weather would be fair. In either case, the trip would take three days. The American commander, General George Kenney, had a choice: concentrate the bulk of his reconnaissance aircraft on the southern or northern route. Once the convoy was sighted, his forces would attempt to destroy the Japanese convoy. In making their decision, the students were asked to employ the concepts defined above.

The student groups were asked: (1) who are the players? That was obvious, they are General Kenney and the Japanese commander; (2) what are the *interdependent choices* available to the players? Here it was for General Kenney to perform reconnaissance north or south of the island, and the Japanese commander to travel north or south of the island; and (3) what are the *utility preferences* for the players? Not all of the students got this correct, but the correct answer is that this is "directly tied to the number of days of bombing (of the Japanese fleet by the allied fleet)."

This ends up being a zero sum game – for the allies, the more days of bombing the better, for the Japanese, the less days the better.

Students were asked to create what is known as a "normal form game grid." This is in Figure 1. In the grid, "the combination of Kenney N, Japanese S, represents 2 days of bombing. This has a utility preference of 2 for Kenney and -2 for the Japanese. Kenney's gain was exactly the Japanese's loss. Figure 2 showed what game theory teaches us, and this was presented to the students after they had performed the exercise. It showed that the optimal result is that Kenney should perform Reconnaissance North, guaranteeing two days of bombing, which was the *expected value* of the game. This meant that the Japanese were in a disadvantage – they couldn't win, and the best they can do is minimize the maximum damage, called the "minimax strategy." However, Kenney pursued the strategy that maximized the minimum damage, called the "maximin strategy." Since this was a true story, we know what happened – game theory predicted it correctly – Kenney placed the bulk of his reconnaissance force to the north, the Japanese sailed to the north, and they encountered two days of bombing. For Kenney, performing the reconnaissance north of the island is known as a *dominated strategy*, because it *ensured* that the Japanese would encounter a maximum days of bombing. For the Japanese, traveling north was their *dominated strategy*, because traveling south might have incurred even more days of bombing than traveling north.

		Japanese N	Japanese S
Kenney	N	2	2
	S	1	3

Figure 1: Normal Form Game Grid for Battle of the Bismarck Sea. Hamman, S., (2018), Lesson 2, slide 21)

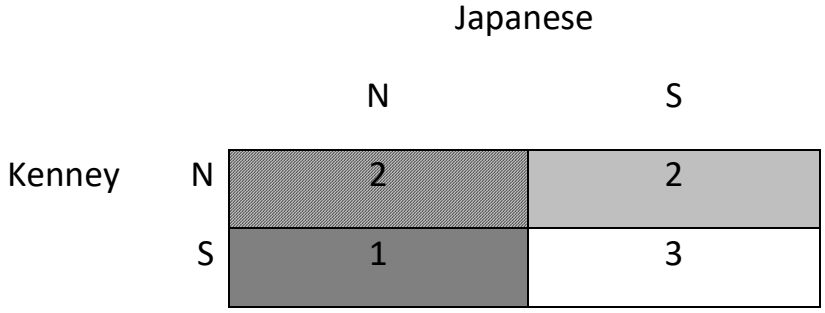

Figure 2: Kenney's dominating strategy is North row, Japanese dominating strategy is North column, and the optimal result for both "players" is cross-hatched intersection. Hamman, S., (2018), Lesson 2, slide 22)

The importance of this exercise is that although students applied game theory to a military scenario, the same lesson can be applied to defending a computer network. For example, knowing a vulnerability in a computer network, determining the optimal location (dominating strategy) to place an intrusion detection system in that network.

Lesson 3 – Introduction to Behavioral Game Theory

The final lesson teaches students how to apply what is known as "level-*k* reasoning" to create playing strategies in strategic contests; the learning outcome is that students will be able to "analyze cybersecurity from the strategic perspective of cyber adversaries."

In Lesson 3, it is stated that "one of the underlying assumptions of game theory is *player perfect rationality,* meaning that "players behave perfectly rationally to the nth degree when making strategic choices." (Hamman, 2018, Lesson 3 slides, slide 5) However, this is not always the case, and a simple exercise illustrated the difference between *analytical game theory* and *behavioral game theory*.

To illustrate this point, the students performed an individual exercise known as the 2/3s Guessing Game. In this game, each student was asked to write down a whole number between 0 and 100, inclusive. Students were told not to look at anyone else's number, and not to share their number. Once everyone was done writing down their numbers, I collected the submissions, did a couple of calculations using Excel, and announced the winner. The winner was the person who submitted the number closest to 2/3 of the average of all of the numbers submitted.

This game is related to behavioral game theory for two reasons. First, it is a theoretical game because it has: (1) players (the students); (2) *interdependent choices* – the whole numbers between 0 and 100; and (3) *utility preferences* – losing and winning.

Second, "the analysis of this game depends on the concept of *dominated strategies,* learned in Lesson 2." (Hamman, 2018, Lesson 3 slides, slide 7) After calculating the answer, the highest possible winning number could only be 67, which is 2/3 of 100, but that would depend on each student choosing 100 as their answer. This means that "all of the numbers between 68 and 100 are *dominated strategies,* and should never be chosen under any circumstances." Consequently, it is necessary to re-do the game in the light of this. So now what's the highest possible winning number? This would result in everyone choosing 67, and 2/3 of that is 45. This means that now all of the numbers between 45 and 67 are also dominated strategies and should never be chosen. So not only can you see where this is going, but where would this process stop? (Hamman, 2018, Lesson 3 slides, slide 8)

The process itself is called the "*successive elimination of dominated strategies*." With *analytical game theory,* this process would "continue all the way to the bottom." This demonstrates "*player perfect rationality,* that all the players would use a strict, logical analysis of the game." (Hamman, 2018, Lesson 3 slides, slide 9) But people don't really think that way. In other games in the module, there are only "1 or 2 successive elimination of dominated strategies, and then the equilibrium is reached." (Hamman, 2018, Lesson 3 slides, slide 9) The goal for the player is to find the optimal number of iterations of the game before the player quits. Each iteration is called a level, and this is called level-*k* reasoning. Most people "engage in between 0 and 3 levels of level-*k* reasoning," and "most of the time, the level-2 strategy is a winner." (Hamman, 2018, Lesson 3 slides, slide 15)

How can level-*k* reasoning be applied to cybersecurity? The adversary finds a vulnerability in your network, you discover that he has discovered it, so you protect it. Then he finds another vulnerability, so you protect that. As defenders, cybersecurity professionals must determine how many layers of security, how many levels of security, they are willing to implement and pay for, to defend against each potential adversary and mitigate each potential vulnerability. At some point, the cost of defending against a possible exploit with a very low probability of occurring becomes prohibitive, and thus we teach our students that is the point where the strategy of acceptance, accepting the very minor possibility of an attack, should be employed.

The lesson closed with yet another exercise illustrating another important issue in adversarial thinking – the *allocation of scarce resources.* Entitled "DDoS Attack," students were given six websites, numbered 1 through 6, each with a value of 1. Students were given 120 protection units. The students allocated their 120 protection units against a set strategy that was unknown when they made their allocations. The set strategy of 2-31-31-31-23-2 was unveiled, and students scored themselves against it. The literature stated that it would be unlikely that most students didn't win at least four of the six "battlefields," and that was the case. The reason can be explained in the three takeaways from this exercise: (1) the end websites were assumed to be "undervalued," so the set strategy won them cheaply against the students. This is called *focal point bias*; (2) level-*k* reasoning was employed in the choices of 31 and 23 in the middle, just in case a student would allocate their units equally (20 on each "battlefield"), so level-*k* reasoning is germane when you have multiple dimensions, and we have that in cybersecurity; and (3) if you let level-*k* reasoning go too deep, you are overthinking the problem. Understanding the allocation of resources leads to the Data Breach Exercise, which was used as the pre- and posttest of this module. (Hamman, 2018, Lesson 3 slides, slides 22-23)

THE DATA BREACH EXERCISE, PRETEST AND POSTTEST

This exercise is very similar to the DDoS Attack exercise described above, but the scenario was somewhat different. The students were given this exercise as a pretest, and again, at the end of the module, as a posttest. So they did have to identify themselves on their answer sheets. But in no way were their answers graded. As was the case for all of the exercises, the students were given a pass-fail grade indicating attendance. They either were present, or they were not. Students were not allowed to makeup the group exercises, as they had to be performed in a group. Students were allowed to make-up any individual exercises that they missed.

Description of the Data Breach Exercise

The scenario describes a large company using an old, but deep-rooted mainframe computer used to collect new customer data. Being as old as it is, the mainframe cannot be properly secured, so every weekend the company runs a large migration job that clears out the data off the mainframe and moves it to a secure server. The company is concerned that an insider might copy all of the customer data off the mainframe and sell it on the black market. Although they cannot technically prevent this, they regularly audit the log files. In the future, they will allocate 100 man hours per week to the task of auditing mainframe's daily logs.

The company collects about the same amount of data each day, so the database grows linearly throughout the week. The database starts fresh every Monday morning because of the weekend job that migrates data to the server. Assume that the number of hours allocated to inspecting a

particular day's jobs equals the likelihood of detecting an attack on that day. So if X hours are assigned to reviewing a day's logs, and an insider attacks on that day, then the probability of detecting the insider is X %. We also should assume that if the insider is detected, the threat will be eliminated resulting in a "reward" equal to 10 points for the company. Each student has been hired as a cybersecurity consultant, and the job of each is to allocate the 100 man hours over the five daily log files, ensuring that the integer percentages all add up to 100. Each student then had to describe their reasoning. While students who missed either the pre- or posttest exercises were offered the opportunity to make them up (within two days of the exercise), if any student had only done the pre- or posttest and not both, the exercise that they did perform was thrown out.

Statistical Results, and Their Meaning

The authors provided detailed instructions for scoring the exercise. It was scored against actual data that they had collected from 33 computer science undergraduate students who had participated in the in the role of attackers. Those students were tasked with selecting the one day of the week that they would attack (unlike the defenders who were tasked with allocating log inspecting hours across days. The results from the attack students are provided in Figure 3, and are considered the control set for the exercise. When viewing Figure 3, you can see that about one half of the attackers chose Wednesday, and about 1/3 chose Tuesday, about 1/6 chose Thursday, and none chose Monday or Friday.

	Monday	Tuesday	Wednesday	Thursday	Friday
Percent of Attacks	0%	36%	46%	18%	0%
Value of Day	1	2	3	4	5

Figure 3: Actual data used as a control to measure student class student data against. Hamman, S., (2018), Data Breach Exercise – Scoring_v10.docx

The formula given for scoring the defender's submissions is in Figure 4. The first half of the formula accumulates points in proportion to detecting an attack on a particular day. In this exercise, the reward was stated as 10 points. The second half of the formula deducts points in proportion to their likelihood of *not* detecting an attack on a particular day. The final score is the sum of these values over all five days. The authors give examples of a possible set of student allocations, and the calculation of that student's points per day.

$$\sum_{i=1}^{|Days|} a_i(d_i * R) + a_i((1 - d_i) * -v_i)$$

Figure 4: Formula for scoring the defender's submissions. Hamman, S., (2018), Data Breach Exercise – Scoring_v10.docx

In the formula in Figure 4: a_i = percentage of attackers who choose day i; d_i = percentage of hours allocated by the defender on day i; v_i = value of day i; R = reward for detecting the attacker.

In the example given in Figure 5, the total of the five days is -0.292. In the Excel workbook they provided, the raw scores are normalized to values between 0 (minimum) and 100 (maximum). So in this example, the student's final score would be 42.3. These scores are not to be interpreted as a percentage grade, but "higher scores do indicate stronger adversarial thinking abilities." When

comparing a student's pretest to posttest scores, "the difference provides an indication as to how the student's adversarial thinking abilities have changed over the course of the module." (Hamman, 2018, Data Breach Exercise – Scoring_v.10, p. 2-3)

Example of student allocation of hours and resulting points per day:

	Monday	Tuesday	Wednesday	Thursday	Friday
Percent of Hours	16%	18%	20%	22%	24%
Value of Day	1	2	3	4	5
	Monday	Tuesday	Wednesday	Thursday	Friday
Points per Day	0	0.0576	-0.184	-0.1656	0

Figure 5: Example of student allocation of hours and resulting points per day. Hamman, S., (2018), Data Breach Exercise – Scoring_v10.docx.

To give an idea of what actual student data looks like, the pre- and posttest data for the first three students (alphabetically) in the class is shown in Figure 6. Only the first three students' data is displayed because it would have been unwieldy to display the resulting data for all twenty-one students.

	PRE /	Hours submitted					Total	Points earned per day					Total	Total
Student	POST	Mon	Tue	Wed	Thu	Fri	Hours	Mon	Tue	Wed	Thu	Fri	Points (Raw)	Points (Adj)
Student A	PRE	5	10	10	25	50	100	0	-0.29	-0.78	-0.09	0	-1.16	27.8
Student A	POST	3	10	16	21	50	100	0	-0.29	-0.42	-0.19	0	-0.902	32.1
Student B	PRE	30	10	20	10	30	100	0	-0.29	-0.18	-0.47	0	-0.94	31.4
Student B	POST	5	30	30	30	5	100	0	0.576	0.414	0.036	0	1.026	64.3
Student C	PRE	25	20	10	20	25	100	0	0.144	-0.78	-0.22	0	-0.854	32.9
Student C	POST	0	12	21	32	35	100	0	-0.2	-0.12	0.086	0	-0.2394	43.2

Figure 6: Actual pre- and posttest data for the first three students in the class

In order to analyze the entire class, a sample t-test was run in Excel, using the T.TEST function. In order to ensure that analysis of the data had no effect on grades, entry of each student's pre- and posttest data was entered into the provided spreadsheet after the course had been completed and grades had been posted. The data confirmed that overall, the students' adversarial thinking abilities had improved. With an N = 19, the mean score pretest was 35.6. The mean posttest score was 52.4. 16 out of 19, or 84.2% of the class, saw an improvement in their score. The *p-value* of the two-tailed t-test of paired pre-post scores was 0.001261, indicating that the results were statistically significant, and the improvement in mean score pre- and posttest was as expected.

SURVEY

In addition to calculating the class' pretest – posttest data, an anonymous online survey was conducted. Although voluntary, since this was an anonymous survey, the enticement to participate was that if all of the class participated, everyone would receive 10 points extra credit. Sixteen students did participate, but unfortunately for the class, that was not 100% of the students enrolled. Nonetheless, I am including the results of the survey. The survey consisted of seventeen statements, with the responses in the range of Strongly Agree to Strongly Disagree. While some of the questions relate to the learning outcomes, several of the seventeen questions related

specifically to the various exercises, but only the questions related to the exercises discussed in this paper have been included, along with the more general questions.

Survey Question	Percent Responding Strongly Agree / Agree
After being exposed to Lesson 1, I feel that I understand the overall concept of Adversarial Thinking and how it applies to hacking.	87.5%
I understand Sternberg's Triarchic Theory (related to cognitive psychology) and how it helps me understand how hackers think.	81.3%
After completing Lesson 1, I feel that I understand the role of strategic reasoning in adversarial thinking	81.3%
The Battle of the Bismarck Sea exercise helped me understand the basics of Game Theory	75.0%
The various exercises in Lesson 2 on Game Theory helped me understand the role of best response analysis (putting yourself in the shoes of your adversary) in strategic reasoning.	81.3%
Participating in the 2/3 Guessing Game exercise helped me understand the concept of successive elimination of dominated strategies	62.5%
Participating in the 2/3 Guessing Game exercise helped me understand level-k reasoning as it applies to Behavioral Game theory	56.3%
Performing the DDoS exercise helped me understand the concepts of strategic resource allocation and level-k reasoning	68.8%
I feel that the entire Adversarial Thinking module was beneficial to my understanding of how a hacker thinks, and how to defend against a hacker.	87.8%
I feel that the time spent on Adversarial Thinking, versus the time that would have been spent on studying Network Security, was worth the inclusion of Adversarial Thinking in the course	68.8%

Table 1: Student Survey Results

CONCLUSIONS

From the statistical analysis of the Data Breach Exercise pre- and posttest results, it is clear that the short module in adversarial thinking described in this paper improved the students' adversarial thinking abilities – they were more able to think like a hacker, and thus able to understand how to allocate defensive assets to prevent attacks on an organization. In addition, the results of the anonymous survey showed that the students felt that they benefited from the adversarial thinking module. In their own literature, the authors indicate that this module can be taught to any level class in cybersecurity, but that it should be taught in a classroom, face-to-face. The optimal course for this module at our university would be our 4000-level Ethical Hacking course, but that is usually taught online. Consequently, since our introductory course in cybersecurity, open to all students, is usually taught in a classroom environment, it is likely that is the course in which this module will be taught in the future. Regardless of which course that this module is taught in the future, the pretest – posttest and the survey will be administered to collect more data in order to further validate these results.

REFERENCES

Clark Center. (n.d.) Retrieved January 7, 2019 from https://clark.center

Hamman, S. (2018), Adversarial Thinking (entire module of lessons, includes PowerPoint slides, exercise descriptions, and exercise scoring examples). Retrieved from https://clark.center/details/shamman/Adversarial%20Thinking

Hamman, S (2018). Adversarial Thinking Lesson 1: Intro to adversarial thinking [PowerPoint slides]. Retrieved from https://clark.center/details/shamman/Adversarial%20Thinking

Hamman, S (2018). Adversarial Thinking Lesson 2: Intro to game theory [PowerPoint slides]. Retrieved from https://clark.center/details/shamman/Adversarial%20Thinking

Hamman, S (2018). Adversarial Thinking Lesson 3: Intro to behavioral game theory [PowerPoint slides]. Retrieved from https://clark.center/details/shamman/Adversarial%20Thinking

Hamman, S. & Hopkinson, K. (2016). Teaching adversarial thinking for cybersecurity. *Journal for the Colloquium for Information Systems Security Education (CISSE),* September 2016.

Kim, D., & Solomon, M. (2018). *Fundamentals of information systems security* (3rd ed., p.79). Burlington, MA: Jones & Bartlett Learning.

Levine, D. What is Game Theory? (n.d.). Retrieved July 29, 2019, from http://www.dklevine.com/general/whatis.htm

Oryiano, S, & Solomon, M. (2020). *Hacker techniques, tools, and incident handling* (3rd ed., pp. 6-7). Burlington, MA: Jones & Bartlett Learning.

NSA CAE-CD 2019 Knowledge Units (n.d.). Retrieved July 28, 2019, from https://www.iad.gov/NIETP/documents/Requirements/CAE-CD_2019_Knowledge_Units.pdf

Stewart, J. (2014). *Network security, firewalls, and VPNs* (2nd ed., p. 113). Burlington, MA: Jones & Bartlett Learning.

The Prisoner's Dilemma (n.d.). Retrieved July 29, 2019, from https://en.wikipedia.org/wii/Prisoner%27s_dilemma