Artificial Intelligence and Game Theory Models for Defending Critical Networks with Cyber Deception

Sunny Fugate, Kimberly Ferguson-Walter

■ *Traditional cyber security techniques have led to an asymmetric disadvantage for defenders. The defender must detect all possible threats at all times from all attackers and defend all systems against all possible exploitation. In contrast, an attacker needs only to find a single path to the defender's critical information. In this article, we discuss how this asymmetry can be rebalanced using cyber deception to change the attacker's perception of the network environment, and lead attackers to false beliefs about which systems contain critical information or are critical to a defender's computing infrastructure. We introduce game theory concepts and models to represent and reason over the use of cyber deception by the defender and the effect it has on attacker perception. Finally, we discuss techniques for combining artificial intelligence algorithms with game theory models to estimate hidden states of the attacker using feedback through payoffs to learn how best to defend the system using cyber deception. It is our opinion that adaptive cyber deception is a necessary component of future information systems and networks. The techniques we present can simultaneously decrease the risks and impacts suffered by defenders and dramatically increase the costs and risks of detection for attackers. Such techniques are likely to play a pivotal role in defending national and international security concerns.*

There have been many recent advances in artificial intelligence and machine learning that have addressed the speed and accuracy of detecting malicious activity so as to better defend networks. Many of these solutions are able to take predetermined actions against detected activities in order to bolster defense and to prevent system compromises — such as dynamically reconfiguring a firewall rule to block attempted denial of service attacks. Today's solutions however, generally stop short of either directly interfering with malicious activity or gracefully responding to more subtle indicators of malicious intent. The current risk is that false

ISSN 0738-4602

positive rates can be quite high and responding to these would cause more harm than good. Often, it is still a human operator who weeds through the alerts generated by an ML-based detector, determines which are true and which are false positives, and then coordinates with a cyber defender to respond to the suspicious activity. The problem is further exacerbated by the limited set of responses available to human and automated protection systems. Today's intrusion detection and intrusion prevention systems include heavy-handed response options such as block suspicious IP or quarantine machine. The limited set of available responses tends to drive both the need for higher-confidence alerts and the necessity of human-in-the-loop decision-making.

In this article, we will show how a handful of network-based interference and packet-manipulation techniques can be combined with game theory and rule-based reasoning to automatically react and respond to attacker activity even when overall confidence in detected events is low. Our current work focuses on a set of goal-driven cyber deception techniques that enable a much richer set of prestaged defensive postures and adaptive responses as well as high-confidence indicators of malicious intent. This emerging area of research is based on well-known principles of human behavior and cognition, which tend to diverge from traditional computer security responses and risk management (Gutzwiller et al. 2018).

Attacker Advantage

To frame our problem statement in a clear manner, we present a notional model of what we mean when we say that the attacker has an asymmetric advantage over a defender. As the attacker's probability of a successful attack increases, the defender's risk of compromise increases and the attacker's risk of detection decreases. In a naive model of risk, these two may be inversely proportional to one another, as shown in figure 1. A slight increase in the probability of success of an attack (including avoidance of detection) would always result in a proportional decrease in risk for the attacker and increase in risk for the defender. This relationship describes, in essence, how cyber risk is estimated today, using linear combinations of weighted measures such as the number of vulnerabilities, the likely severity of a breach, and the cost of mitigation or recovery. However, this model fails to account for asymmetries in cyber warfare as it exists today. The attacker is often not detectable using traditional pattern-based detection schemes and may often also be able to choose one of many possible vulnerabilities and many possible systems to exploit. This imbalance results in a situation more in line with that depicted in figure 2. The problem is made worse by the techniques cyber attackers have long used, such as deception, distraction, and the use of previously unknown, and therefore indefensible, attacks. The defender's risk dramatically increases as the probability of a successful attack increases. Attackers expend few resources, while defenders bankrupt themselves mitigating vulnerabilities and recovering from breaches. The area between the curves represents the differences in risk between attackers and defenders. When an attack has a low likelihood of success, the defender has an advantage. When an attack is likely to succeed (given all defensive techniques known to the defender), the situation is reversed.

While these models are only notional, they provide a means for reasoning about what type of change particular defenses are likely to bring to a cyber defense scenario.

Game Theory

Unlike many games of conflict that are studied in the academic literature, computer security games tend to have some unique properties that make them both interesting and somewhat challenging to model (Roy et al. 2010). Games such as chess and Go, while computationally challenging from a state-space perspective, have many properties that lend themselves well to formalization in a game theory framework. In such games, each player has the same set of available strategies and the same overall goal, making the game symmetric and the players interchangeable. Players are given generic and interchangeable identifiers, such as Player 1 and Player 2. Network and computer security games, however, have players who have asymmetric strategy sets and goals that are not only opposing, but that might have very different end states and resulting payoffs. In general, we treat one player as the defender and the opposing player as the attacker. This distinction is important, as the players in these types of games are not interchangeable.

Fortunately, many security games can be modeled around a common resource relating to one of the well-known computer security measures of confidentiality, integrity, or availability (the often-cited CIA triad). In cases where players are aware of one another's goals, they are likely to have game outcomes that can be modeled in a zero-sum fashion. For example, a defender's goal might be to prevent data exfiltration, whereas the attacker's goal might be to achieve data exfiltration. Each player is reasoning about the confidentiality of information, but with opposing goals. Any successful exfiltration of information by an attacker represents loss of confidentiality for the defender. While the actual value (and assigned weights) of the information might differ between players, we can set this aside for later discussion.

In other cases, the defender and attacker might have goals whose end states are not easily comparable. This situation occurs when some aspect of the

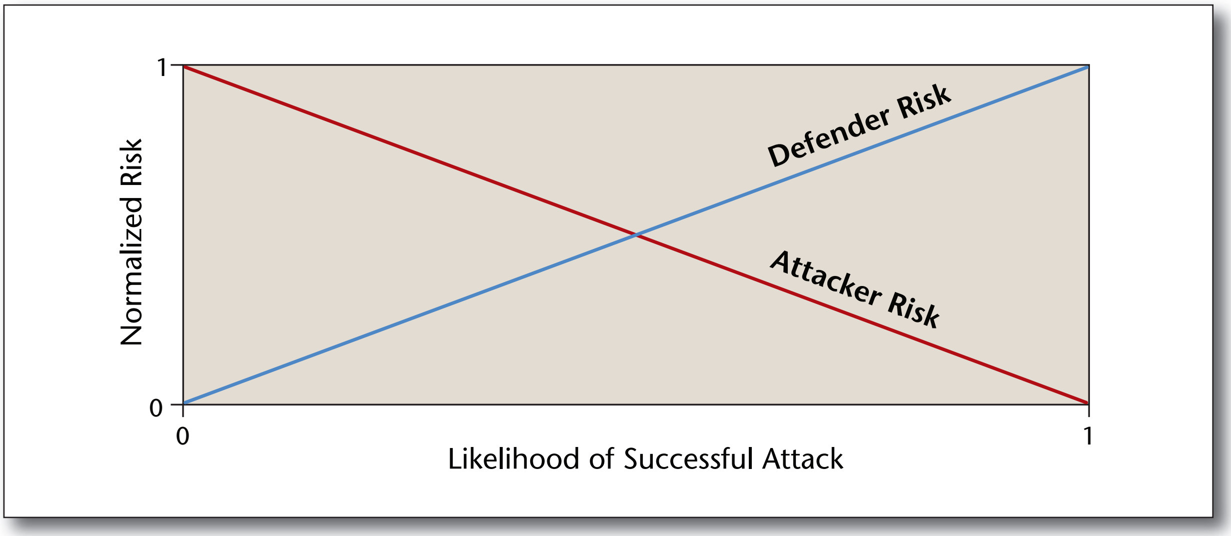

Figure 1. Naive Model.

In a naive model, the attacker and defender risk curves are inversely proportional. As the probability of a successful attack increases, attacker risk decreases and defender risk gradually increases.

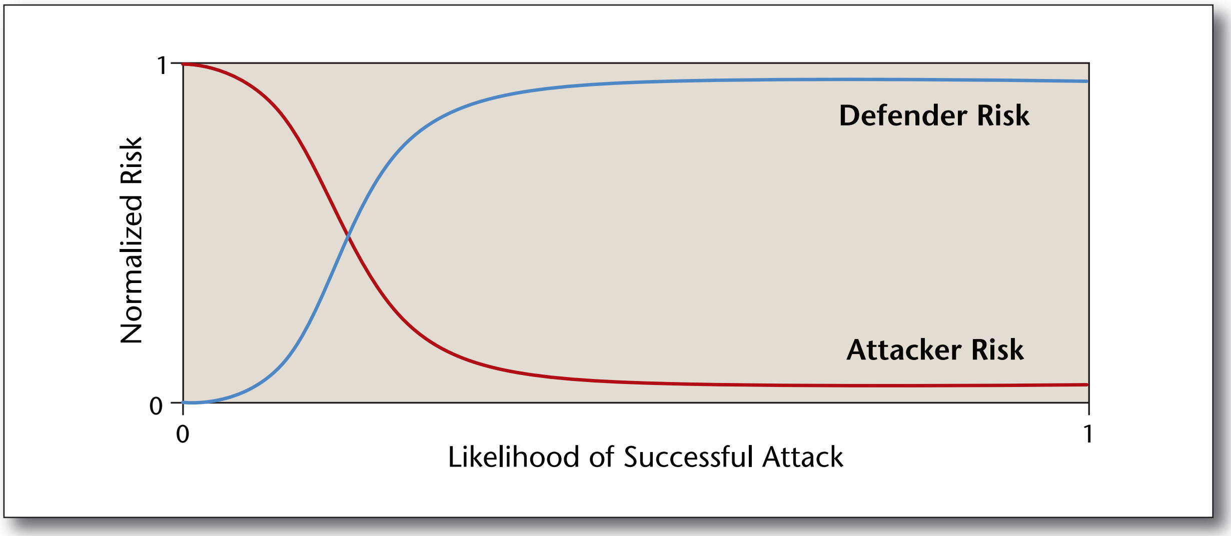

Figure 2. Asymmetries Cause Model to Fail.

All too often in cyber warfare, the attacker has an asymmetric advantage over the defender through the use of deception, distraction, novel exploits and cost asymmetries (such as the low cost of exploiting multiple systems and the high cost of remediation).

game is hidden from one or both players. For example, an attacker might desire to exfiltrate data, whereas the defender might be solely concerned with perimeter defense, with a goal of denying any and all access from attackers while allowing unfettered access to legitimate users. This kind of scenario is common in network security situations and results from the defender being unaware of the attacker's goal. As a result, the defender often uses a defense that might be exceptionally effective at preventing initial access to systems on their network, but that is quite poor at preventing data exfiltration for connections that are already established or that are created by hosts residing on the inside of the corporate firewall.

Traditional game theory uses the ideas of incomplete and imperfect information to describe scenarios where one of the players has incomplete knowledge of the sets of strategies available or the payoffs of game end states (for either themselves or the other player) or imperfect knowledge of the history of moves taken. Some card games, such as poker or blackjack, deal cards to each player over several rounds. The specific history of cards that are dealt (or held, for blackjack) can be only partially known from the player's own hand or by those played face up (by counting card frequencies as they show up in play). In such games, the players are symmetric with respect to goals, available moves, and how much information each player has about the history of the game. Each player knows their own hand (and history) perfectly, but must attempt to infer the cards played or held by others by taking into account those shown to all players. Deception and bluffing also play a role in these games, which we will use as a springboard for discussing cyber deception techniques shortly.

In such games, the payoff value of particular cards might also be unknown. In a game such as Texas hold'em poker or even go fish, players each have information which is shared to the group of players, but for which the player's actual payoff values for each displayed card are unknown. In go fish, players request cards from others, possibly indicating a preference for a particular card. In Texas hold'em, the community cards are visible to all players and a player's willingness to continue the game can also indicate information to their opponent(s). In either game, players may bluff and request cards they do not need, or continue play even though their hand is actually poor. As we will show later, both learning preferences and bluffing are desireable strategies in computer security environments and one of the outstanding challenges that we have been attempting to solve through our research on defensive cyber deception.

Another complexity of real-world situations such as those appearing in cyber defense is one of ethics. In most situations, we must intentionally restrict defender behaviors based on legal and ethical principles. Unfortunately, attackers do not necessarily have such restrictions. While an ethical defender is generally not allowed to harm network-based attackers, attackers often exploit computer networks specifically for the purpose of harming the defender or the defender's systems. As noted earlier, this imbalance presents a situation that provides the attacker with significant and unfair advantage. Such advantage often makes game-theoretic analysis of formal models of the behaviors somewhat uninteresting — the defender always loses. In game theory terms, analysis becomes much easier (and sometimes trivial) when there exist strongly dominated strategies for which no move is beneficial for defenders. It is our intent in this paper (and in our research) to demonstrate feasible techniques to both hinder the attacker and improve the defender's advantage, so as to make these games more interesting and to even tip the balance in the direction of the defender.

Cyber Deception

Traditional security controls tend either to require being built into the systems a priori or to focus on blocking known malicious behavior or new suspicious behavior. These security controls, while important, have limited successes. Security solutions focused on hardening systems and perimeter defense are missing the dynamic component that deals with new events that have thwarted our current defenses and gained access to our network/systems. There are always new vulnerabilities, new tactics, new ways to gain illegitimate access. Some solutions focus on detecting anomalous behavior and then either quarantining it or ejecting it from the network. While this solution might temporarily handle a security breach, ejecting a persistent attacker will just lead them to find a new strategy. This cycle can continue until they have found a path that is unprotected and they are undetected. So, it can be argued that just blocking suspicious activity can lead in the long term to more undetectable attacks on the network. This disadvantage is precisely why the cyber defender notoriously has the harder job. An attacker need only find one way in. Is it even possible for an operational network to be constantly protected at all points?

The cyber defender's job is difficult, but defensive cyber deception is an emerging area of research that might bring some advantage back to the defender (Heckman et al. 2015; Rowe and Rrushi 2016). Cyber deception can be used to delay, misinform, and deter a cyber attacker. Such deception will at least slow down the attacker's successes, and in the best case, misinformation can be used to disrupt or deter an attack altogether. There are a variety of cyber deception techniques that can delay an attacker, including decoy systems, tarpitting, and honeypots. Cyber deception can also strategically provide misinformation about the network to a potential attacker. If the attacker does not know the true network topology or the types of systems and services running, launching a successful attack becomes much harder. Currently, the networks we are trying to defend are providing the attackers with true information, which the attackers are then using to attack and harm those same networks. Yet, anyone who has broken through the initial security barrier of a network should be treated as nefarious, with no right to accurate information or good usability of the systems. Cyber deception, like encryption, is a good tactic for obscuring our critical information from attackers who wish to steal and abuse it. Attackers rely on observation of digital information for their intelligence and generally have very limited ways to corroborate the information they are presented. For this reason, deception is a powerful tool for cyber defense, and, we argue, should be used as ubiquitously as encryption.

Since cyber deception is based on principles of human cognition, it has the potential to affect the attacker rather than just the attack. This refined targeting allows for lasting effects — effects that can disrupt future steps in the attack chain and future attacks by that attacker. By providing confusing and incorrect information about key terrain the attacker needs to complete their goals, we are able to do more than block one step in the attack; we are able to potentially disrupt or deter that attack (and maybe future attacks) altogether.

Cyber Deception Games and Hypergames

One of the simplest types of games we can analyze is one in which an attacker can choose whether or not to attack and a defender can choose to defend (or detect) or not to defend. In the simplest form of this game, there is only one system to detect and the player payoffs are symmetric and zero-sum. Figure 3 shows this game in normal form, where each player's strategies — that is, the choice between whether to take action (to attack or defend respectively) or not — are depicted as α or ᾱ respectively. This simple game is meant to reflect the kinds of choices made by attackers and defenders. A defender might choose not to defend for a round because of the potentially high cost of defending. Skipping a round saves resources. An attacker might choose not to attack due to the cost of attacking or because they believe that the system is defended and their attack will fail, resulting in them being ejected from the network.

In the diagram, if the attacker chooses to take an action, denoted by α, then they are attacking. If the defender chooses strategy ᾱ, then they are not taking an action (and not defending). While this game's structure is too simple to embody the nuances of cyber deception, it will serve as a springboard for later discussion. It should be noted that while we will describe each player as *attacker* and *defender,* in the simplest form of this game the players are essentially interchangeable. A notable exception is that the payoffs for ᾱ are opposing. The attacker has an advantage for attacking in the top-right and bottom-left quadrants. The defender wins only in the top-left situation, again illustrating the asymmetric advantage enjoyed by an attacker.

In this game model, the attacker's goal is either to successfully attack an undefended system or (for defended systems) to cause the defender to incur a cost for defending the system even when it is not being attacked. The defender's purpose is to defend against an active attack, while not wasting resources in attempting to defend against an attacker who hasn't yet attacked. The defender has limited computing resources and desires to ensure that each defense is useful. While this model and the player's respective goals are gross simplifications, some of the realities of cyber attack and defense are nonetheless reflected. Defenders tend to be resource constrained, whereas attackers tend to desire to remain undetected. This kind of interaction is a common scenario on actual computer networks.

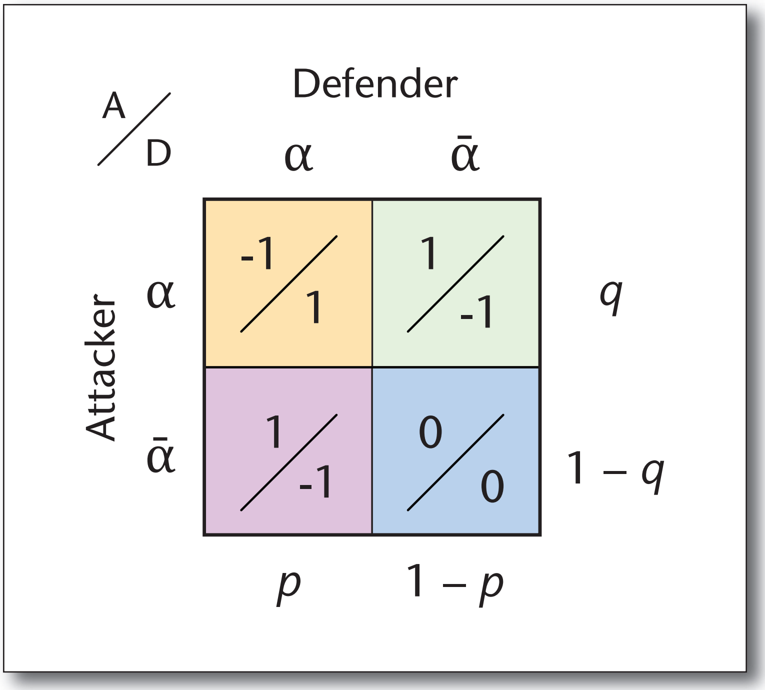

Figure 3. A Simple Defense Game Payoff Matrix for a Zero-Sum Game of Complete Information.

In this game, the attacker is penalized for attacking while being detected and rewarded for attacking while not being detected. The defender is penalized for detecting while not being attacked and rewarded for detecting while being attacked. The logic here is that an attacker wants either to attack undetected or to cause the defender to waste resources, and the defender wants to detect attacks and not to use resources needlessly.

The analysis of this type of game is straightforward and consists of finding equilibrium strategies that meet some predefined notion of optimality. In many cases where the players are in conflict and not cooperating in any way, the most-often used equilibrium is called the *Nash equilibrium* after Nobel Laureate John Forbes Nash (Nash 1951). In a Nash equilibrium (if one exists), neither player has an incentive to unilaterally change their strategies. This analysis for the game in figure 3 can be performed via simple inspection of the game, which results in players always switching strategies. In this game model, there is no state in which one of the players would not choose to change their decision and get a better payoff. That is, if we assume we are currently in any given quadrant, one of the players will choose to switch strategies for a higher payoff. As a result, at least in this game formulation with its specific payoffs, there are no pure Nash equilibrium strategies. This type of analysis assumes perfect information where both players know ahead of time precisely what the strategy of the other player will be. Games of imperfect information are also possible and we will return to this later.

Since the game has no pure Nash equilibrium, we can set up an equilibrium expression to find a mixed strategy that allows players to choose each individual strategy according to a probability distribution across their choices. Because this is a zero-sum game, there always exists a mixed Nash equilibrium that is straightforward to compute using von Neumann's minimax theorem (von Neumann 1928). If the attacker chooses to attack with probability p, the expected utilities for the defender are $D_{\alpha} : (1)p + (–1)(1 – p) = 2p – 1$ and $D_{\bar{\alpha}} : (–1)p + (0)(1 – p) = –p$. The attacker can minimize the maximum payoff of the defender when $2p – 1 = –p$, so $p = 1/3$. Similarly, if the defender chooses to defend with probability q, then the expected utilities of the attacker are $A_{\alpha} : (–1)q + (1)(1 – q) = 1 – 2q$ and $A_{\bar{\alpha}} : (1)q + (0)(1 – q) = q$. The defender can minimize the maximum payoff of the attacker when $1 – 2q = q$, so $q = 1/3$.

To anyone familiar with basic game theory, this result is hardly surprising. However, it suggests that we should reflect on our current strategies for performing cyber defense. The game described assumes very little about the differences between attacker and defender rewards and penalties. It suggests that if we know little about the penalty for not defending, the cost of defending, or the payoffs of the attacker (other than their sign), then we should enable our defense only a fraction of the time. This type of analysis has been performed by a myriad of researchers on a number of elegant games related to cyber defense (Fugate 2012). However, their applicability to real-world scenarios is limited. The most egregious limitation of such models is that the game structure and payoffs are mostly contrived (primarily, but not always, for the purposes of making the analysis of the game interesting). So, while such games provide interesting insights, they tend to lack realism.

This same simple game can also be used to describe instances where the defense strategy is to perform a defensive cyber deception action such as replacing a real system with a decoy. Without other constraints or stipulations, such a strategy would also follow the mixed strategy of enabling the defense with a probability of 1/3. However, this assessment assumes perfect and complete information on the part of both players. In this simple game, the attacker is fully cognizant of the strategy employed by the defender and knows of the existence of the defensive strategy and the probability of its being enabled. In our research endeavors, we are calling an attacker who is aware of deception *sophisticated* (as opposed to *naive*). Interestingly, when our game model includes deception, in addition to attackers having only partial knowledge of defender strategies, an attacker might also suffer from false knowledge (a belief in something that is untrue). That is, if a deception technique is successful, the attacker will not only have partial information regarding the defender's strategy (which in this case is knowledge of the mixed strategy probabilities of the defender), but might also suffer the effects of false knowledge about the environment.

For this adjustment to be modeled, allowing for the existence of false or partial knowledge on the part of the attacker, we must introduce additional game theory concepts: extensive form game representations, Stackelberg models, and the use of suboptimal play as a defender strategy over a sequence of game rounds. We introduce these concepts in the next section and extend our initial game and its analysis directly.

A Model for Cyber Deception

One of the characteristics of cyber environments is that the actions of each player tend to be triggered by the actions of the other player. The player who takes the first action often (but not always) has an advantage (the *first mover advantage*). In traditional cyber defense scenarios, the defender waits until an attacker makes a (detectable) move and then responds. This strategy results in a situation in which the attacker has the advantage and the defender must clean up and deal with the repercussions of the attack. A fundamental goal of our research is to reverse this situation. In our cyber deception game formulations, we attempt to create a situation in which the defender makes the first move by prestaging deception technologies on a network and selecting how these technologies will be deployed. This strategy allows the defender to control the initial environment, and as we shall see, to control the likelihood of attacker success.

In the prior section, we analyzed a simple defense game presented in normal form, appearing as a matrix of player strategies with each box showing payoffs for each player. A game with n attacker strategies and m defender strategies would have $n \times m$ game scenarios with a pair of payoffs for each. Games with more than two players can also be modeled this way. However, games in which a sequence of actions is taken by players in turn (such as chess) is more usefully modeled in extensive form (see figure 4). In this representation, a tree is drawn that represents each stage in the game, with each level of the tree representing the possible choices that can be made by one of the two players. Similar to a normal form game representation, each branch of the tree concludes in a leaf in which payoff values are specified for each player.

Games that have a leader-follower structure, such as the extensive form game in figure 4, are also called *Stackelberg games* after the German economist Heinrich Freiherr von Stackelberg (von Stackelberg et al. 2011). A key aspect of these kinds of game models is that the leader commits to their strategy for a particular subgame based on backward induction, accounting also for the decision that the other player will make as a response. This type of game is no longer played in a simultaneous fashion, as is the simple normal form game. What makes the extensive form game difficult to analyze is that we must optimize the leader's strategy against all possible follower responses. For many realistic cyber defense scenarios, we might have long chains of player actions and responses, and in some games, chess for example, the state space of such a game grows very quickly.

Both extensive form representations and Stackelberg game formulations also align with our goals of modeling information hidden from the attacker. The first player's move might be entirely nonobservable to the attacker, as, for example, when the defender deploys decoy systems that look identical to real systems on the same network. In other cases, the attacker might be able to partially observe the defender's choices (or have prior knowledge of likely moves). In these cases, the attacker will use their knowledge of the defender's move to optimize their own choices.

Rounding out our deception game model, we must also introduce the concept of repeated games in which play progresses through many independent rounds. Attackers might make only a single attempt at reconnaissance or exploitation on a network, but more often than not will make many attempts, often sending hundreds to thousands of packets to dozens of machines while remaining essentially undetected. This strategy is particularly true of network-based reconnaissance, where an attacker sends packets that occur frequently and that are perceived as innocuous but that lead to extensive knowledge about the network and potential vulnerabilities. Each packet sent by an attacker can be considered by a deceiving defender as a single action in a cyber deception game. If the defending system's responses are fast enough, then responses can be made on a packet-by-packet basis. If the defender is adjusting and prestaging defensive deception, then they are able to take the role of leader in a Stackelberg game. Over many rounds, the defender can use their first mover advantage to win more rounds than the attacker or, as we will see in the next section, use deception to lure the attacker into believing the defender is a poor opponent engaged in suboptimal play.

The Value of Suboptimal Play

For repeated games, the value of suboptimal play is illustrated in (Bilinski, Gabrys, and Mauger 2018) by mathematically showing the disadvantage that defenders suffer when using a rational greedy strategy. We argue that by using cyber deception, the defender can create an illusion of playing suboptimally, which presents an opportunity to shift the advantage away from an attacker.

The types of cyber games that we are concerned with also concern human psychology, which includes our ability to reason about nonrational players. A perfectly rational player is generally easier to model. The specific mechanisms of nonrationality are often difficult to pin down and can vary between individual players, making any kind of formal analysis quite difficult. Even when players are perfectly rational, seemingly simple games such as the prisoner's dilemma (Poundstone 1992) have been shown to have many somewhat unexpected results when players use a longer-term strategy over a repeated game. To those unfamiliar with it, the prisoner's dilemma deals with a scenario in which two convicts are provided the option of decreasing their sentence if they defect and rat out the other prisoner. The game is interesting in that if the players were to cooperate, they would have a higher overall utility, but if either player defects, then they should both defect, resulting in a pure Nash equilibrium strategy that is worse than the social optimum. For repeated versions of the prisoner's dilemma, various strategy policies can be defined that take into account the entire history of prior plays. *Grim trigger* and *tit for tat* are two strategies that are often studied for repeated versions of the prisoner's dilemma. The grim trigger strategy uses a policy of cooperating forever unless the other player defects and then always defecting afterward. Tit for tat defects if the other player defects and cooperates if the other player cooperates. In tournaments in which strategies are tested against one another, strategies can be better or worse depending on which

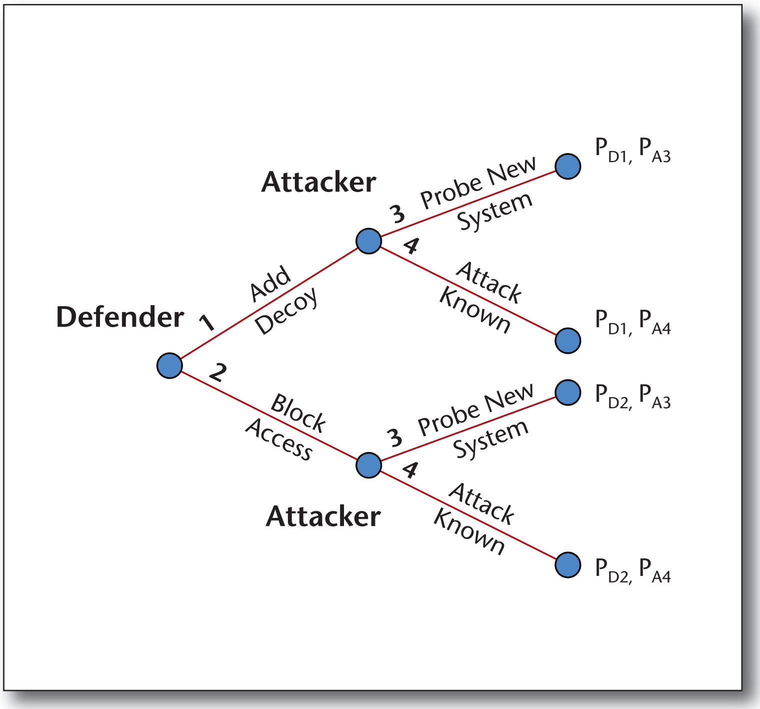

Figure 4. Example Cyber Deception Game Tree in Extensive Form.

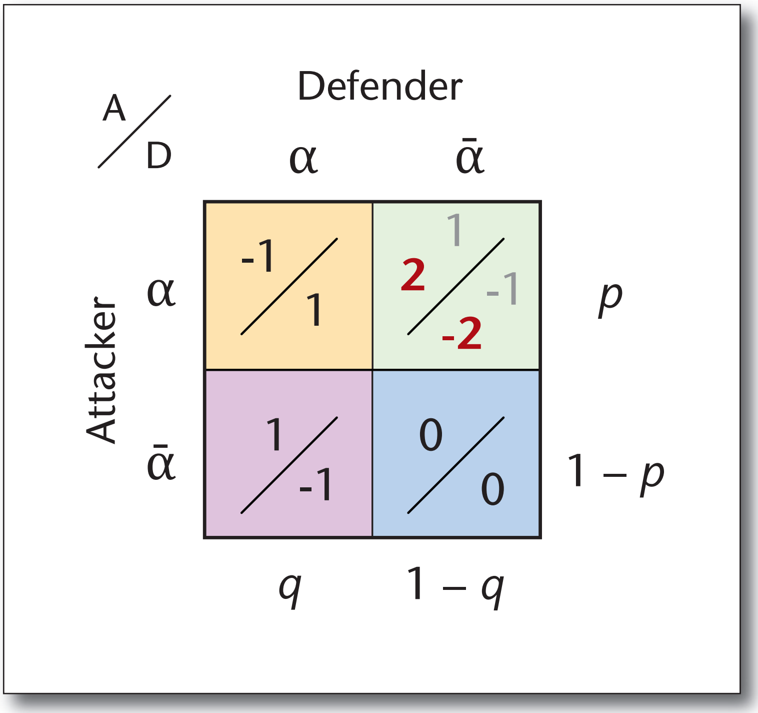

Figure 5. The Defense Game.

In the upper-right quadrant, false payoffs are in red and true payoffs are in gray.

strategy the other player chooses or whether players switch between strategy policies over time (Axelrod and Hamilton 1981).

What is relevant to our research, and to our discussion of games of cyber deception, is that nonrationality can be thought of as a strategy policy over a repeated game. One such policy that we have been exploring is one in which a human player suffers from an expectation bias. In this scenario, a player makes observations of the choices of the other player that seem to represent consistent behavior and then matches their expectations of that player with a particular type. For example, an attacker might perform an assessment of a particular service port on a system that is often used for system management or administration (as with SSH, for example). Having seen several systems with this port open, they might then assume the same configuration will be present on all of the systems on the same network. Similarly, the attacker might observe behavior by systems and assume that this prior behavior will continue without change, predicting that the systems will behave consistently. A defender can construct a situation where they play suboptimally, perhaps keeping a service port that is used for system management open without filtering, allowing an attacker to connect to this port, and displaying service banners that indicate a poor security posture (such as using an outdated and insecure version of the management software). This kind of situation occurs frequently on networks and is exactly the type of vulnerability that attackers seek when looking for opportunities to compromise systems. Over many repeated games, our goal is for an attacker to have an overall low utility. If the defender always plays optimally (for example, by not presenting insecure services), then in many cases we might be forced into maintaining a Nash equilibrium strategy. For many situations, choosing a Nash equilibrium strategy might be better for the attacker than the defender. In our analysis of games of deception we believe that this imbalance is primarily due to the asymmetries present in cyber environments. When defensive deception is not present attackers need only successfully attack a single system, whereas a defender must provide adequate defensive measures for as many systems as possible on a continuous basis. There are also asymmetries in how defenders and attackers reveal information, with a defender's systems often naively providing state and version information to all who request the information.

A key insight of our work is that deception allows defenders to create scenarios that cause an attacker to believe they are choosing an optimal strategy (or a Nash strategy), when in fact, due to defender manipulations of the environment, they are being lured or goaded into choosing a suboptimal strategy. If we modify the payoffs of the defense game in figure 3, we can end up in such a situation. Figure 5 shows our original defense game with a false set of payoffs in the upper-right quadrant (shown in red).

In this model of the game, the defender's payoffs are adjusted to simulate a high cost of failing to defend (an intentional ruse that presents false values to the attacker, shown in red in the payoff matrix). As before, analysis of this game is straightforward. If the attacker chooses to attack with probability p, the expected utilities for the defender are $D_\alpha : (1)p + (–1)(1 – p) = 2p – 1$ and $D_{\bar{\alpha}} : (–2)p + (0)(1 – p) = –2p$. The attacker can minimize the maximum payoff of the defender when $2p – 1 = –2p$, so $p = 1/4$. If the defender truly had the red payoff in the upper-right quadrant, and if the defender chooses to defend with probability q, then the expected utilities of the attacker would be $A_\alpha : (–1)q + (2)(1 – q) = 2 – 3q$ and $A_{\bar{\alpha}} : (1)q + (0)(1 – q) = q$. The defender would then minimize the maximum payoff of the attacker when $2 – 3q = q$, so $q = 1/2$.

In this formulation of the game, a defender having the upper-right payoff of 2/–2 would need to defend more frequently (with a probability of 1/2) and the attacker attack less frequently (with a probability of 1/4). However, the premise of the game is a false one, perpetrated by the defender, who has caused the attacker to believe they had new payoffs in the upper-right quadrant. With this deception, a defender can make an attacker believe that there is a higher cost for unsuccessful defense (that is, a higher-value target than the true system values). This misrepresentation is exactly what is done in traditional honeypot techniques — simulation of high-value targets that actually have very low value. In this game, the true payoffs have not changed and are still 1/–1 as in our original game model. The attacker has analyzed the game and chosen (what they believe to be) the new Nash equilibrium strategy. The attacker's new probability of attacking is now fixed at $p = 1/4$. This is good for the defender. Fewer attacks are occurring, but the defender is free to defend with the original strategy of 1/3 instead of 1/2. The defender now has an advantage due to the decrease in attack frequency. Furthermore, the attacker believes they are playing the Nash strategy and that they can do no better.

It is important to note that falling back on the original strategy is a safe strategy for the defender to take. Even if the attacker becomes aware of the deceptive payoffs, the defender cannot do worse than the original game. However, if the defender is confident in their deception, they might also adjust their strategy, potentially decreasing their rate of defense to better take advantage of the decrease in attack frequency. We leave computation of the new optimal defender strategy, given the fixed attacker strategy, as an exercise for the reader.

A common practical example of where this analysis is important is when a network is completely secured against known threats. The attacker will seek novel defects or misconfigurations, which will often be unknown and undetectable to the defender. However, if the defender presents a suboptimal strategy, one that presents service ports and versions that are vulnerable, the attacker will take the lower-cost strategy of attacking the known vulnerability. The suboptimal play by the defender lures the attacker into making a false payoff prediction and making a decision to commit to using a greedy strategy. Assuming the attacker has taken the bait, the defender can use deception to continue the ruse while performing adjustments to the network or to the behavior of systems with which the attacker is interacting. Cyber deception techniques can make the initial suboptimal play (having a vulnerable service, for example) just an illusion, and thus as safe as any other type of defense.

Deception Hypergames

Hypergame theory is an extension of game theory that is particularly applicable to games of cyber deception. A hypergame is a complex game in which at least one player has a misperception about the model of the game being played. In a hypergame, players might (a) be unaware that they are playing the game, and (b) be unaware of the possible moves in the game (Kovach, Gibson, and Lamont 2015). The attacker might not even know a cyber deception game is being played, and even if made aware of the certainty of deception, would not know what types of deceptive moves were available to the defender. In a cyber deception game, the defender's game tree might look very different from that of the attacker, and the hypergame model can encompass all of the subgame trees as they are played out for each individual player's perception of the game. Further discussion, game tree examples, and formal notation for modeling cyber deception as hypergames is presented in a paper by Ferguson-Walter et al. (2018).

Manipulating the Gameboard

Cyber deception is a powerful tool for defenders because it allows them to manipulate the gameboard, which has traditionally been a possibility only for attackers. We believe that the use of deception itself is a primary cause of the current asymmetry of cyber warfare. However, as the owners of the network, cyber defenders should be able to control the information the network distributes and potentially change the way the network behaves. Such control would be akin to the defender changing the gameboard in the midst of a game of conflict with the attacker. In our estimation, this type of game manipulation is able to give the defender an asymmetric advantage over an attacker. The gameboard can be manipulated in several ways, which can have various effects on the attacker.

By changing the gameboard that the attacker sees, the defender is able to limit the strategies available. If the attacker has the wrong information about a system, the strategies they think are applicable to attack will likely fail. Additionally, as noted, the hypergame model can encompass both the manipulations of the gameboard and the nonrational strategy policy used by the defender.

One major advantage that cyber deception provides to a defender is the ability to change the perceived payoff to the attacker. Each player is selecting actions and trying to maximize a long-term payoff. The payoff is an estimation of how good or bad the outcome is for that player. Recall that many game theory games are structured as zero-sum games, where the payoffs for each outcome add up to zero across the players.

Since the defender can control the information the attacker uses to make their decisions (and form their game tree), the defender can manipulate the payoffs that the attacker associates with certain paths. For example, a defender can make a system look more vulnerable or more interesting. This distortion will cause the attacker's perceived payoff for that machine to be much higher than the true payoff. Furthermore, if the defender is using decoys or honeypots, the attacker's perceived payoff might be very high, while the true payoff is instead very high for the defender. This negative true payoff for the attacker is due to the time and energy wasted on a fake system, which is evident in human subjects studies on the effects of cyber deception (Ferguson-Walter, LaFon, and Shade 2017).

Learning in Cyber Deception Games

For a defender to make wise decisions about how to best protect their network and systems, there are several useful things they need to know. First and foremost, the defender will be more effective if they know when they are being attacked. They can use proactive defenses including preset cyber deception techniques, but the effect will be greater if they can also adapt those defenses based on details of a current attack in real time. In addition to knowing that an attack is occurring, knowing details about the attacker and their actions will also help them develop a better defense. Learning the preferences of the attacker (for example, they tend to attack Linux machines), the attitudes of the attacker (for example, they are noisy and not careful to avoid detection), and patterns of behavior (for example, the attacks occur at certain times of day) will aid the defender in customizing the gameboard and launching the best cyber deception.

This learning cannot be completed through typical supervised learning classification techniques. There is no existing dataset with labels and information about the best tactic for these cyber defense situations. This is also not an unsupervised learning clustering problem. To select the best response at any time, it does not help us to look for things that are similar and then to group them together. Semisupervised learning, such as reinforcement learning, is likely the best tactic for this problem space. As the defender observes the actions of the attacker and interacts with them, the estimates and probabilities in the defender's model of the world will be created and updated. Feedback to know whether a tactic has been successful or not is a critical, but complicated, component of semisupervised learning.

Reinforcement Learning

In general, reinforcement learning algorithms are a class of adaptive control algorithms that, through repeated interactions with a controlled system, learn to optimize some function of the state of that system. The system has the state set X and the action set A. Executing action a from state x causes a transition to state y with probability $P(x, y, a)$. At each time step, t, the controller selects an action a_t based on observation and estimation of the current state, x_t. The system executes a_t, resulting in a state transition to x_{t+1}. The reinforcement signal from the previous time step, r_{t-1}, is used to adapt the controller over time so as to optimize a function of the sequence of reinforcement signals (Sutton and Barto 1998).

A policy, or a control policy, is a function that takes some representation of the current system state as input and generates an action to take, thereby inducing a change to the system state. Typically, the action is chosen so as to meet certain desirable criteria. Policy constraints are parameters, boundaries, and strategic guidelines that steer the selection of an acceptable policy. Policy constraints place restrictions on the types of actions that are available to the policy in different situations.

Although RL algorithms have been used to build adaptive, optimizing controllers for many different kinds of systems, the theoretical foundations of RL assume that the system to be controlled is a Markov decision process (MDP). In an MDP, the system is a controlled Markov chain, where the state transition probabilities depend only upon the current state and the chosen action. Moreover, despite our best efforts to instrument the network to gather the necessary sensor data, there will be components of the system state that cannot be directly observed. This imperfect visibility means that we will be working with a partially observable Markov decision process (POMDP) and that we will have to adopt some means of estimating enough of the hidden state for the system to learn and improve over time (Spaan 2012).

The game theory models provide the RL algorithm with a reward signal or expected payoff for various actions. The algorithm can also provide a path to estimating the hidden state of the POMDP. The defender's game tree details all the possible paths and action sequences. This detailed mapping allows the defender to map from an observation to a path on the tree, thus estimating the current state, potential payoffs, and appropriate action.

Finding the optimal policy for a cyber defense scenario is complicated by the fact that this is inherently a multiobjective optimization problem: at each time step, the RL algorithm is presented not with a single cost, but with a vector of costs and rewards. For example, to find a good policy, the RL algorithm must find a way to equitably trade off a number of goals — such as (1) minimize communication costs, (2) minimize computation costs, (3) minimize disruption to defended systems, (4) maximize system availability, (5) minimize sensor costs — and then potentially invert these measures for the purposes of disrupting an attacker or causing an attacker to incur long delays or high communication or computation costs.

Model Extensions

Currently our model considers only games consisting of two game trees (Ferguson-Walter et al. 2018). A future extension of our work would expand this model to consist of additional game trees for various types of attackers and defenders. Our model currently does a reasonably good job of intuitively describing a player's model of the opposing player's choices. However, when both players have uncertainty and when either might be deceived, we now must quadruple the number of game trees — from a single tree used in traditional extensive form games to four trees — so as to describe each player's own model as well as each player's model of the opposing player.

Extending this logic, we must also be capable of dealing with situations of counterdeception. In the context of this paper, counterdeception refers to a situation where the attacker is also using deception against the defender.

Nonomniscient Defender

In our initial model, we make the simplifying assumption that one of the two players has complete and perfect information concerning not only their own model of the game, but also that of their attacker. The assumption of an omniscient defender implies that the defender knows all potential attacker strategies, actions, costs, and payoffs. This assumption is based on the idea that through the use of deception and manipulation of the cyber environment (such as swapping out a real machine for a decoy), a defender knows the environment because they fully control it. Missing from this model are characteristics and strategies of the attacker that might be unknown to the defender. For cyber environments, this information is particularly relevant. Deception techniques are often used by attackers to manipulate perceived goals and to misdirect defender resources, for example, by using a denial of service attack to mask more subtle activities and to keep defenders preoccupied in recovering systems to acceptable service levels.

Further, attackers use assumptions of safety and security to break the rules that systems are built on. If the defender were truly omniscient, then the system and all of its underlying assumptions would be perfectly modeled and the defender would be aware of all possible attacker actions, strategies, costs, and payoffs. Attackers rely on this very assumption to ensure that if they have high-confidence knowledge that their actions are currently undetectable (such as through the use of a zero-day exploit), then the defender will not be capable of using defenses against those strategies. Indeed, zero-day exploits represent situations in which attackers know of a defect that is new and that defenders might not be capable of recognizing or detecting. Today, attackers are correct in their assumptions: current defense techniques forego taking actions against unobservable attacks. While it would seem like a logical impossibility for a defender to take an action against an unobservable attack, this is precisely what the defender does when we prestage deception to interfere with unknown attackers and attacker actions. In our opinion, it is not only possible to model feasible strategies against unknown and unobservable attackers, but it is necessary if we are to be in any way capable of improving the status quo.

While our current model does not directly address these concerns, it does allow for explicit representations of unobservable moves. Each layer in our game tree includes an additional branch to represent undetected or unobserved moves by each player, and each player can use Bayesian reasoning to base new moves on the likelihood of an unobserved (and thus unknown) action being taken.

In the context of our cyber deception games, from the defender's perspective, the most important unobserved move by an attacker is any interaction with a real system when there is no indication of maliciousness. In a situation such as this, we have reverted to the default cybersecurity situation where defensive deception is not present. Cyber deception can mitigate this situation to an extent, but our model currently assumes that real and decoy are perfectly indistinguishable to an attacker. In reality, there might exist observable signals that could cause an attacker to be more interested in real systems or there might be defects in the deployment of decoys that deanonymizes them.

From the attacker's perspective, the most important unobserved moves by defenders consist of cyber deception actions where the attacker's knowledge of the game environment is completely undermined and replaced with an alternate reality. Similarly, the most important attacker strategies from a defender's perspective are those in which deception has little or no effect. When defensive deception strategies are absent, the defender will always be at more of a disadvantage than when they are in use. In an environment in which attackers and defenders are both using deception, neither player has complete information about the game, but both suffer the consequences of deception. The defender in such a game cares most about attackers for which the deception has no effect, and attackers care most about defenders who are effectively using deception techniques.

The asymmetric nature of unobserved and unobservable actions in cyber deception games (and models of cyber games in general) is a fundamental part of military deception strategies as they have been understood historically (Whaley 1969). Incorporating these concepts into cyber defense is a natural extension of prior work and provides ample opportunities for future research in the application of game theory to the defense of computing systems.

Conclusions

Effective cyber defense must incorporate a mixture of "security hygiene" (for example, patching systems) and dynamic adaptive techniques. Adaptation is something that human attackers do very well. It is a critical component of why and how they often manage to thwart current security measures. If the attacker can quickly and easily adapt their attack and the defenders cannot do the same with their defense, then the defenders are playing a losing game. An adaptable security technique must be able to change quickly and to evolve over time if it is to match wits with the attackers. This autonomy is a component of cybersecurity well suited to an AI solution.

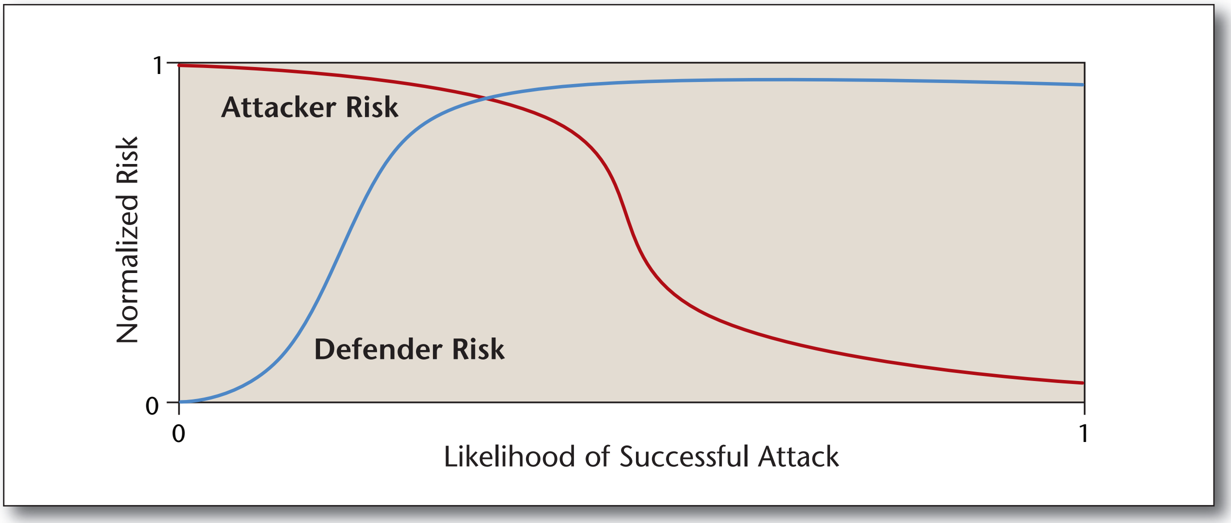

Figure 6. Traditional Defenses.

Improving traditional defenses increases attacker risk due to increased likelihood of detection and increased costs for attacks.

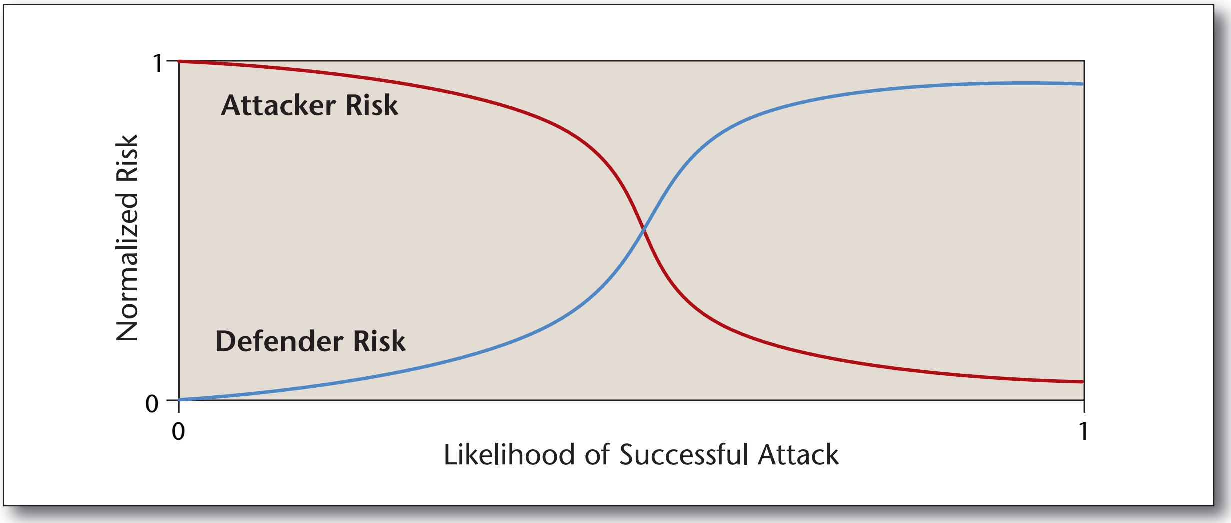

Figure 7. Resilience Mechanisms.

If the defender incorporates additional resilience mechanisms, then defender risk also decreases.

In order for an AI solution to adapt effectively, it must not only learn about the actions of the attacker, it must also have a mechanism for receiving feedback on its own decisions and actions and for updating its estimates accordingly. The game theory model discussed here begins to provide a framework for representing and updating estimates of likely strategies and outcomes in the form of changing game tree structures and payoff scores. Using game theory to represent the conflict between the attacker and defender, to model the difference in perception, and to reason about the best course of action can be a critical component of an adaptive cyber deception system.

The AI defender must attempt to infer the attacker's beliefs over time and apply them to its decisionmaking. As the defender receives observations from the environment of the attacker's activity, the defender will need to use this information to model the state of the attacker in its game tree and to estimate the attacker's perceived payoffs. With knowledge of the current game tree and evaluation of likely attacker beliefs, the defender can now autonomously select a response that will manipulate

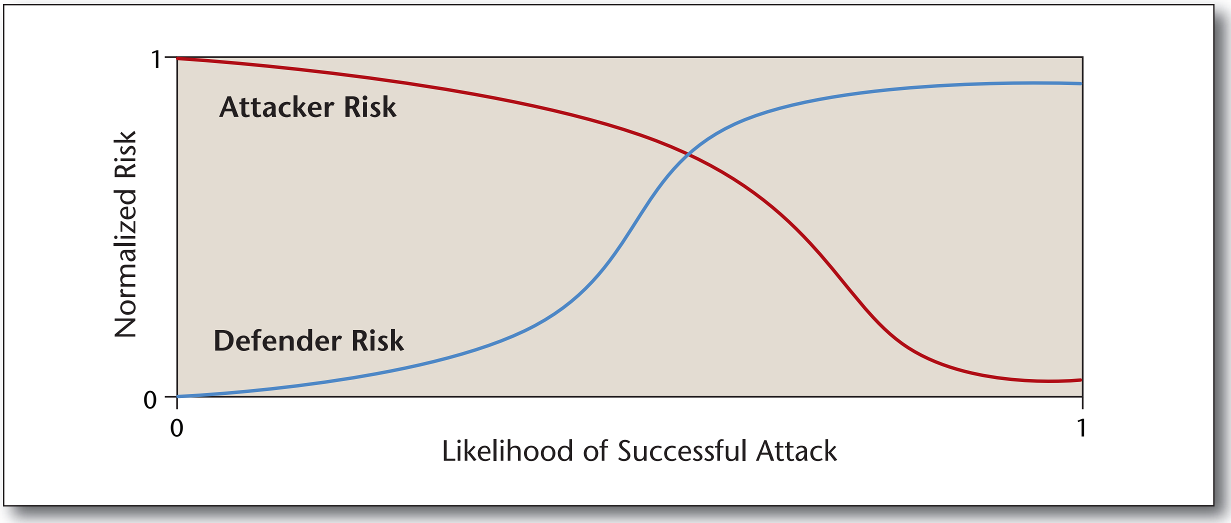

Figure 8. Deception.

A defender using deception can gain an advantage over the attacker, greatly increasing the risk to attackers due to detection or even retaliation by defenders.

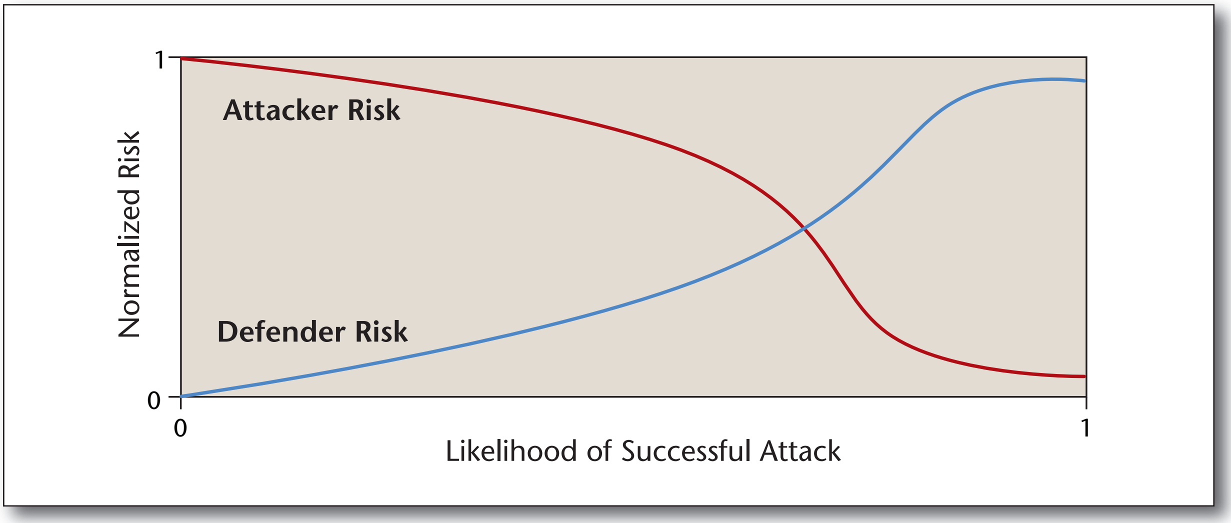

Figure 9. Counter Deception.

A defender using counterdeception also properly contends with deceiving attackers, further decreasing defender risk.

the gameboard to change the actual payoffs associated with the next possible actions. This is an iterative process where the defender must continuously learn about the attacker through observation and update its models and evolve its strategies accordingly. The decisions made and deceptive actions taken by the defender both manipulate the payoffs the attacker can receive and limit the strategies available to the attacker at the next time step. This kind of autonomous reasoning for cyber deception can generalize to autonomous cyber defense, though both are still in the early stages of research.

In conclusion, we have described how traditional game theory can be extended to provide practical guidance for cyber defense scenarios. We have also provided arguments for why a cyber defender should be making use of cyber deception as a principle strategy for defending systems. We framed our problem statement with a discussion of the asymmetric advantage that attackers enjoy in traditional cyber environments. We argue that the asymmetry arises not only from the legal and the ethical challenges of defenders, but also from a reticence to use deception in the implementation of defensive strategies. Finally, we present a notional view of how we believe that cyber deception can improve defender advantage. Figures 5–8 describe the effects of various strategies: improving traditional defense mechanisms (figure 5), incorporating resilience (figure 6), employing defensive cyber deception (figure 7), and finally using counterdeception to thwart attacker deceptions (figure 8). The mechanisms in figures 5 and 6 are already common tools used by defenders, but they get us only part of the way to shifting the advantage to defenders. In our perspective, cyber deception and counterdeception are key elements of a successful cyber defense strategy and necessary for any well-reasoned approach to the defense of networks and networked systems. To force attackers to suffer the same disadvantage as defenders do, we must employ many of the same basic techniques. Just as the attacker's goals, their techniques, or their very existence is often unknown to a defender, we must make our critical information systems equally opaque.

References

Axelrod, R., and Hamilton, W. D. 1981. The Evolution of Cooperation. *Science* 211(4489): 1390–96. doi.org/10.1126/science.7466396.

Bilinski, M.; Gabrys, R.; and Mauger, J. 2018. Optimal Placement of Honeypots for Network Defense. Paper presented at the Conference on Decision and Game Theory for Security. Seattle, WA, October 29–31.

Ferguson-Walter, K. J.; Fugate, S.; Mauger, J.; and Major, M. 2018. *Game Theory for Adaptive Defensive Cyber Deception*. Technical Report 3141. San Diego, CA: US Navy SPAWAR Systems Center Pacific.

Ferguson-Walter, K. J.; LaFon, D. S.; and Shade, T. B. 2017. Friend or Faux: Deception for Cyber Defense. *Journal of Information Warfare* 16(2): 28–42.

Fugate, S. 2012. Methods for Speculatively Bootstrapping Better Intrusion Detection System Performance. PhD dissertation, Department of Computer Science, University of New Mexico, Albuquerque, NM.

Gutzwiller, R.; Ferguson-Walter, K.; Fugate, S.; and Rogers, A. 2018. "Oh, Look, a Butterfly!" A Framework for Distracting Attackers to Improve Cyber Defense. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* 62(1): 272–76. doi.org/10.1177/1541931218621063.

Heckman, K. E.; Stech, F. J.; Thomas, R. K.; Schmoker, B.; and Tsow, A. W. 2015. *Cyber Denial, Deception and Counter Deception: A Framework for Supporting Active Cyber Defense*. Advances in Information Security 64. Cham: Springer. doi.org/10.1007/978-3-319-25133-2.

Kovach, N. S.; Gibson, A. S.; and Lamont, G. B. 2015. Hypergame Theory: A Model for Conflict, Misperception, and Deception. *Game Theory* 2015(2): 1–20. doi.org/10.1155/2015/570639.

Nash, J. F. 1951. Non-Cooperative Games. In *Annals of Mathematics* 54: 286–95.

Poundstone, W. 1992. *Prisoner's Dilemma*. New York: Doubleday.

Rowe, N. C., and Rrushi, J. 2016. *Introduction to Cyberdeception*. Berlin: Springer. doi.org/10.1007/978-3-319-41187-3.

Roy, S.; Ellis, C.; Shiva, S.; Dasgupta, D.; Shandilya, V.; and Wu, Q. 2010. A Survey of Game Theory as Applied to Network Security. In *43rd Hawaii International Conference on System Sciences*, 1–10. IEEE. doi.org/10.1109/HICSS.2010.35.

Spaan, M. T. J. 2012. Partially Observable Markov Decision Processes. In *Reinforcement Learning*, edited by M. Wiering and M. van Otterlo, 387–414. Adaptation, Learning, and Optimization 12. Berlin: Springer. doi.org/10.1007/978-3-642-27645-3_12.

Sutton, R. S., and Barto, A. G. 1998. *Reinforcement Learning: An Introduction*. Cambridge, MA: The MIT Press.

von Neumann, J. 1928. Zur Theorie der Gexellschaftsspiele. In *Annals of Mathematics* 100: 195–320. doi.org/10.1007/978-3-642-12586-7.

von Stackelberg, H.; Bazin, D.; Urch, L.; and Hill, R. 2011. *Market Structure and Equilibrium.* Berlin: Springer.

Whaley, B. 1969. *Stratagem: Deception and Surprise in War.* Cambridge, MA: Artech House.

Sunny Fugate is a senior research scientist for the US Navy's SPAWAR System Center, Pacific and the center's senior scientific technical manager (SSTM) for cyber warfare. During the last 16 years, Fugate has run numerous research programs to explore the intersections of cyber defense, cognitive science, game theory, and artificial intelligence. Fugate earned a BS in electrical engineering from the University of Nevada in 2002 and a PhD in computer science at the University of New Mexico in 2012. Fugate's current efforts are focused on improving the human factors of cyber defense and exploring opportunities to improve cyber defense using defensive deception and game theory.

Kimberly Ferguson-Walter is a senior research scientist with the US National Security Agency's Information Assurance Research Group. She earned a BS in information and computer science from the University of California, Irvine, and an MS in computer science from the University of Massachusetts, Amherst, both specializing in artificial intelligence. She is currently a PhD candidate at the University of Massachusetts, Amherst, with a focus on adaptive cybersecurity. Her research interests are focused on the intersection of computer science and human behavior. She has been focused on adaptive cybersecurity at the agency for the past eight years and is the lead for the Research Directorate's deception for cyber defense effort. She is currently on joint-duty assignment to SPAWAR Systems Center, Pacific to perform collaborative research and facilitate strategic alignment and technology transfers.