Honeypot Allocation over Attack Graphs in Cyber Deception Games

Ahmed H. Anwar, Charles Kamhoua and Nandi Leslie
US Army Research Lab
2800 Powder Mill Rd, Adelphi, MD 20783
Email: a.h.anwar@knights.ucf.edu, charles.a.kamhoua.civ@mail.mil nandi.o.leslie.ctr@mail.mil

***Abstract*—In this paper, we propose a scalable algorithm to allocate honeypots over an attack graph. We formulate a two-person zero-sum strategic game between the network defender and an attacker. This formulation captures the importance of different nodes inside the network. The game mode accounts for the cost associated with different defense strategies as well as the cost paid by the attacker. Moreover, this game model considers a practical threat model with respect to the available information about the attacker to the network defender. Nash equilibrium defense strategies are analytically characterized and studied for a special game. The complexity of a general game is discussed and a scalable algorithm is proposed to obtain honeypots allocation strategy in large scale networks. Finally, samples of our numerical results are shown to verify our findings.**

I. Introduction

Modern computer networks are highly connected and heterogeneous in order to provide more complicated services and adapt to increasing and rapidly changing demands. For instance, these networks connect computers with different operating systems and protocols. Moreover, an increasing number of devices are being added to networks everyday [1]. For example, the deployment of wireless-enabled devices [e.g., Internet of Things (IoT), robots, sensors] has made networks larger and denser. Hence, networks are prone to higher levels of interference, which makes them more vulnerable. The diversity of devices also makes maintaining them (e.g., patching vulnerabilities) a much more challenging management problem. Such security issues are even more challenging in the military environment.

The IoT network structure is also being deployed in battlefield contexts, where it is known as the Internet of Battlefield Things (IoBT) [2], [3]. In a broader sense, the IoBT also refers to devices useful for military battles that may communicate over tactical networks other than the Internet. Therefore, protecting the resilience and robustness of critical nodes of such a network on the battlefield is crucial. In this paper, we propose an approach for optimizing cyber deception to prevent attackers from characterizing effective attack strategies. It is evident that attackers gather information about the targeted systems/networks before launching their attacks [4]. In the information gathering stage (also called the reconnaissance stage) an attacker collects inside information about the targeted network using a set of tools and scanning techniques [5]. Attackers typically map the targeted network using software scanning tools like Nmap [6] or infer the network via traffic analysis. On the other side, the network administrator (the defender) can effectively protect his network at this early stage of reconnaissance by deceiving the attacker and manipulating the network interfaces to disguise the true state of the network. Specifically, we investigate an attack scenario in which the defender protects critical nodes and important system components via introducing false information (i.e., deception) to disrupt the attacker's decision-making, providing him with a false sense of certainty.

Cyber Deception Games (CDG) are a recent but growing topic in the game theory literature, modeling scenarios in which the defender protects the network by manipulating its state, and in some cases using decoys or deceptive signals [7], [8], [9], [9], [10], [11]. The defender may camouflage nodes to hide critical networked devices. Recent work formulated honeypot deception games as a strategic game [12], [13], [14]. Schlenker et al. [15] proposed a defense approach based on a Stackelberg game model to increase the attacker uncertainty about the system parameters to misguide the network scanning tools. Hence, the attacker gets false information about what operating system is running on what machine, what port is assigned to which service, and the names of subnetworks and active users. In [16] the authors presented a Stackelberg game model in which the defender chooses optimal mitigations that reduce the capability of the attacker to achieve his goals. The authors also investigated complexities and scalability challenges of such problems.

The applications of these sorts of game theoretic models can shape the understanding of robustness for connected devices in a tactical network (e.g., IoBT). Over the past several decades, there has been a growing concern with network resilience for industries and government, given that security breaches are pervasive, and cyber deception can be an effective proactive network resilience approach to this end. For example, in 2019, the U.S. Department of the Army published a document on its strategy for military deception for multi-domain operations [17], including network security, that addresses deception–this includes the presentation of false weakness in information systems–as a means to dissuade network attacks.

To this end, we formulate a two-person zero-sum strategic form game to optimally allocate honeypots over the network and obtain practical deception strategy. Moreover, we characterize Nash equilibrium strategies to both players. After that, we extend the deception game model to consider ℓ levels of

978-1-7281-4905-9/20/$31.00 ©2020 IEEE

deception in the network. The complexity of the extended model is investigated and a scalable algorithm to overcome the exponential complexity of the game is proposed. We present numerical results that validate the proposed defense approach and the efficiency of the scalable solution as well.

This paper is organized as follows: In Section II we present our model for the wireless network, along with the attacker and defender goals. We present our game formulation in Section III and present our scalable approach. We present our numerical results in Section IV and conclude the paper in Section V and discuss ongoing and future research to this work.

II. System and Game Model

In this section, we formulate the game model between the two players, define the control variables of each player and finally discuss the reward function and the game parameters.

A. Attack Graph

We consider an attack graph of N nodes represented as a graph $G(\mathcal{V}, \mathcal{E})$, where $N = |\mathcal{V}|$. Each node represents a vulnerability associated with a host or a machine in the network. An edge $e_{u,v} \in \mathcal{E}$ is connecting two nodes u and v indicates reachability to exploit a vulnerability at node v through a vulnerability at node u. Each node in the graph has a value, w_v, that reflects its importance to the network administrator and hence nodes with a higher value are considered a valuable asset to the network that contains important databases and critical information to the military group. Therefore, it is practical to assume that nodes of high value are very attractive to adversaries and network attackers. The attacker wants to maximize his expected reward through wisely selecting a victim node among the set of all reachable nodes. We start our game formulation and analysis by assuming that the attacker knows the values associated with each node. This assumption is justified since attackers usually can obtain some internal information regarding the network structure and can probe nodes with network scanning tools, [6].

For a simple illustration, Fig. 1, shows a system model and the associated game to be played between the network admin and the attacker. In this figure we have 3 nodes, let for instance node $a \in \mathcal{V}$ represent an entry node to the network. Node a has two edges, $e_{a,b}$ and $e_{a,c}$ connecting it to node b and c, respectively.

We assume that the defender does not know the exact location of the attacker to consider a practical threat model. However, the defender knows that the attacker can possibly penetrate his network through a set of entry points. Since the network records can easily provide a distribution $f_a(.)$ over the set of entry points, where $\mathcal{V}_e \subset \mathcal{V}$ denotes the set of entry points. In other words, the defender know the probability that an attacker will penetrate the network through an entry point $u \in \mathcal{V}_e$ is $f_a(u)$, such that $\sum_{u \in \mathcal{V}_e} f_a(u) = 1$. We can now readily define the game played at each of the entry points.

B. General Game Formulation

A two-player zero-sum strategic game is defined as a triple $(\mathcal{N}, \mathcal{A}, \mathcal{R})$, where,

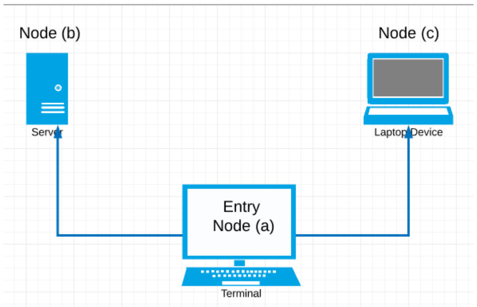

Fig. 1. System model illustration

- $\mathcal{N} = \{1, 2\}$ is the set of players, where player 1 is the network defender and player 2 is considered to be the attacker.
- $\mathcal{A} = \mathcal{A}_1 \times \mathcal{A}_2$ is the game action space, where $\mathcal{A}_1$ and $\mathcal{A}_2$ is the defender and attacker action spaces, respectively. An action profile denotes a joint action taken by the two opponents (a_1, a_2) that determines the reward received by both players.
- $\mathcal{R} = \{\mathcal{R}_1, \mathcal{R}_2\}$, where $R_1 + R_2 = 0$. $\mathcal{R} : \mathcal{A} \rightarrow \mathbb{R}^2$ is the reward function for the defender and $\mathcal{R}_2$ is the attacker reward function.

1) Defender Action: The defender allocates a set of k honeypots along the attacker path to deceive the attacker from reaching his target and mislead his actions. Such honeypots will deviate the attacker from reaching real targeted nodes. Honeypots are allocated as fake services along the different edges connecting between network vulnerabilities. If the attacker exploit such fake service (honeypot) placed over $e_{u,v}$ to reach node v from u, the defender will know the current location of the attacker. Moreover, the fake service will reveal important information about the attacker hacking techniques, etc. This tracking is extremely important for the defender to make his future defense actions. However it is beyond the scope of the paper to study the game dynamics, instead, we focus on maximizing the defender reward due to tracking the attacker at every possible game. Therefore, the proposed solution represents the core of the solution to any dynamic game formulation which is part of our future work.

Assuming that the attacker is starting launching his attack at node a as shown in figure 1, the defender who is allocating one honeypot needs to decide which edge to place the honeypot on. Recall that the defender is not certain about that the attacker is at node a, he also considers a no-allocation action as a possible choice to avoid the allocation cost as discussed below. Overall, the defender action in this case is to place the honeypot on $e_{a,b}$ or $e_{a,c}$ or no-allocation.

2) Attacker Action: On the attacker side, his action is to decide which node to attack next. Given the current example in figure 1, the attacker chooses to exploit the vulnerability at either node b or c. Since the attacker wants to remain stealthy, there is a cost per attack. Hence, the attacker may choose to completely back-off in some circumstances to avoid the attack cost. An attack cost represents the risk of getting caught that is associated with the attack action. Finally, the attack action space is to either attack node b or c or to back-off.

C. Reward Function:

The network defender incurs a fixed cost for placing a new honeypot on an edge in the network. Let P_c denotes the honeypot placement cost. On the attacker side, there is a cost per attack denoted as A_c. The attack cost reflects the risk taken by the attacker as mentioned earlier. If the defender placed honeypot on the same edge the attacker exploits, the defender gains a capturing reward. Otherwise, if the attacker exploited the other safe edge, the attacker gains a successful attack reward. Let Cap and Esc denote the defender *capture* reward and the attacker successful reward (i.e, *escape* reward), respectively.

To account for different nodes in the network, we adopt a reward function that takes into account the importance of the network nodes. Therefore, players' rewards are weighted by the value of the secured or attacked node value, w_v. We start by expressing the reward matrix for the game illustrated in figure 1 and present the general reward matrix afterward.

$$R_a = \begin{bmatrix} -P_c + A_c + Cap * w_b & -P_c + A_c - Esc * w_c & -P_c \\ -P_c + A_c - Esc * w_b & -P_c + A_c + Cap * w_c & -P_c \\ +A_c - Esc * w_b & A_c - Esc * w_c & 0 \end{bmatrix}. \tag{1}$$

The attacker reward matrix, $R_2 = -R - 1$.

The reward function can easily be generalized to an arbitrary number of possible edges as follows.

$$R_1(a_1, a_2) = \begin{cases} -P_c + A_c + Cap * w_v \; ; & a_1 = e_{a,v}, a_2 = v \qquad \forall v \in \mathcal{V} \\ -P_c + A_c + Esc * w_u \; ; & a_1 = e_{a,v}, a_2 = u \forall u \neq v \in \mathcal{V} \\ -P_c \; ; & a_1 = e_{a,v}, a_2 = 0 \qquad \forall v \in \mathcal{V} \\ 0 \; ; & a_1 = 0, a_2 = 0 \end{cases} \tag{2}$$

where $a_1 = 0$ denotes that the defender is not allocating a honeypot and $a_2 = 0$ denotes that the attacker decided to back-off.

D. Mixed Strategy

We have defined the actions available to each player in the game, i.e, $\mathcal{A}_1$ and $\mathcal{A}_2$. A pure strategy is a strategy that selects one of these actions. Alternatively, a player may choose to use a randomized (mixed) strategy defined through a probability distribution over these actions. Given the set of actions of player 1, $\mathcal{A}_1$, let $\Pi(\mathcal{A}_1)$ denote the set of all probability distributions over $\mathcal{A}_1$. Then, the set of mixed strategies for player 1 is $\Pi(\mathcal{A}_1)$, denoted by $\mathcal{X}_1$. Therefore, in a mixed strategy $\mathbf{X} \in \mathcal{X}_1$, action a_1^i is played with probability x_i such that,

$$\mathbf{X} = \left[x_1, x_2, \ldots, x_n\right]^T, \tag{3}$$

where $n = |\mathcal{A}_1|$. Similarly, the attacker may also play a randomized strategy, $\mathbf{Y} = \left[y_1, y_2, \ldots, y_m\right]^T$, where $m = |\mathcal{A}_2|$.

Hence, the expected admin reward, denoted U_1, can be expressed as

$$U_1 = \mathbf{X}^T \mathbf{R_1} \mathbf{Y} \tag{4}$$

Each player aims to maximize his own reward. In a zero-sum game, this implies minimizing the other player's reward. The expected utility for player 1 in equilibrium, $U_1 = -U_2$. The minimax theorem implies that U_1 holds constant in all equilibria and is the same value that player 1 achieves under a minimax strategy by player 2. Using this result, we can construct the optimization problem of player 1 as a linear program (LP) as follows,

$$\begin{aligned} \underset{\mathbf{X}}{\text{maximize}} \quad & U_1 \\ \text{subject to} \quad & \sum_{a_1 \in \mathcal{A}_1} r_1(a_1, a_2) x_{a_1} \geq U_1, \qquad \forall a_2 \in \mathcal{A}_2. \\ & \sum_{i=1}^{n} x_i = 1, \quad x_i \geq 0, \; i = 1, \ldots, n \end{aligned} \tag{5}$$

The first constraint follows from the definition of the Nash Equilibrium. Therefore, the expected reward is greater than the value of the game. Since the value of the game depends on the mixed strategy played by player 2 (the attacker), the admin should constrain his response to the best response set that guarantees a higher reward. The remaining two constraints ensure that $\mathbf{X}$ is a valid probability distribution. The attacker solves a corresponding LP that can be characterized along the same lines to ensures that the optimal mixed strategy $\mathbf{Y}$ is a best response for every possible action played by the defender. The resulting mixed strategies forms a Nash equilibrium for the two players.

Game Complexity: The complexity of the two linear programs grows linearly with the degree of the entry node which is usually a small number compared to the network size. Therefore they can be solved efficiently to obtain the optimal defense and attack mixed strategies (i.e, optimal in Nash equilibrium sense) for both players. However, if the defender is allocating ℓ number of honeypots at the same time to cover a set of edges, the complexity of the LP of such game will be growing exponentially with ℓ. To overcome this complexity, we propose a progressive decomposition algorithm in the following section.

III. ALLOCATING ℓ HONEYPOTS MODEL

In the previous section, the defender tried to defend nodes located one hop away from the entry point node. In this section, we consider a model in which the defender defends nodes located ℓ hops away from the entry node. Since the attacker is taking a path inside the network, the allocated honeypots need to be covering a path in the network as well. Otherwise, allocating ℓ honeypots randomly will not surely secure the set of nodes we are considering in this game model. Moreover, random allocation may result in losing the location of the attacker.

Remark. *In the dynamic version of the game, it will be reasonable to consider allocating ℓ honeypots since the game is evolving and the defender is maximizing his expected future reward. Hence it is acceptable for the defender to lose the location of the attacker in one of the game stages.*

To solve this game we formulate a new LP. However, one needs to enumerate all the possible pure actions for each player as in (5). It is obvious that the complexity of such an optimization problem is growing exponentially in ℓ. To overcome such problem, we propose a progressive decomposition-based algorithm. The analysis of the algorithm performance will appear in an extended version of this paper.

A. The Algorithm:

The reward function of allocating ℓ honeypots is expressed as a sum over ℓ-steps reward functions as follows:

$$\bar{R} = \sum_{h=1}^{\ell} R_1(a_1^h, a_2^h) \;\; ; \; (a_1^h, a_2^h) \in \mathcal{A}_1^h \times \mathcal{A}_2^h. \qquad (6)$$

Note that the action space for both players in the h^{th} hop depends on their action history. Therefore, in order to redefine the action spaces, one needs to keep track of the previous actions taken by both players. This implicit dependency is coupling the objective function of the game LP.

To decouple this dependency the algorithm starts at the network entry point where all the information of the game is available to both players. After the one-hop depth game is played, the resulting mixed strategies are passed forward to the next set of nodes. Hence, the probability of being at each of the nodes that are located 2-hops away from the entry node is known and the new game is now properly defined. In other words, the new action spaces for both players are known. After repeating this procedure ℓ times we multiply the mixed strategies of allocating a honeypot at every edge over all the possible paths and normalize the resulting strategies to have a well-defined probability distribution.

```
Result: Normalize $\mathbf{X}, \mathbf{Y}$
$h = 1$;
while $h \leq \ell$ do
    Define all games $h-1$ away from entry;
    Calculate probability of each game : $P_g$ using $\mathbf{X}^{h-1}$,
      $\mathbf{Y}^{h-1}$ ;
    if $P_g > 0$ then
        Solve LP associated with game $i$;
        Find $\mathbf{X}_i^h$,$Y_i^h$;
        Move to the next game;
    else
        Move to the next game;
    end
    $h++$ ;
    Forward $\mathbf{X}^h$, $\mathbf{Y}^h$.
end
```

Algorithm 1: Progressive decomposition-based algorithm.

IV. Numerical Results

In this section, we present our numerical results and discuss the effect of game parameters as defined in section II.

In Fig. 2 we plot the defender reward at different attack costs. It is obvious that the defender reward increases as the attack cost increases. As the attack cost increases, the rational attacker tends to back-off more often (i.e, with higher probability). Therefore, the reward of the defender increases.

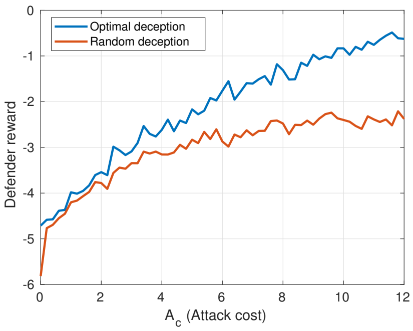

Fig. 2. Comparing defender reward at NE and random allocation at different attack cost

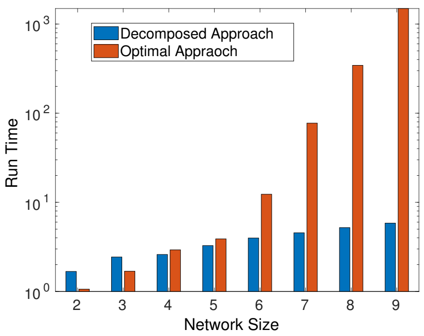

Fig. 3. Algorithm run time

We also compare the reward when the defender adopts NE deception strategy to random allocation strategy to show the effectiveness of our game formalism. The shown comparison is for the network shown in Fig. 1, of only two nodes. As the number of nodes increases, the gap between the optimal reward and random deception reward will increase dramatically. These results are obtained at $Esc = 2, P_c = 5$ and $Cap = 2$. In Fig. 5, we plot the defender reward versus the placement cost, P_c. As shown in the figure, the defender reward decreases as the cost per allocation increases. However, the NE deception strategy is yielding higher reward for the network defender. To show the effectiveness of the proposed decomposition algorithm we compare the run time required to obtain exact random strategy to the decomposition-based randomized strategy in Fig. 3. In Fig. 4, we plot the defender reward on a 7-node network as plotted in Fig. 6. For this network, $\ell = 2$. The algorithm solved one game in the first round and two other games at $h = 2$. The proposed algorithm is yielding a better reward for the defender that is higher than simply randomized the allocation policy blindly. As shown in Fig. 4 the defender reward when the attacker is having stronger computational capabilities and can solve the full game while the defender is using the decomposition-based algorithm solely.

V. Conclusion and Future Work

We proposed a scalable honeypot allocation algorithm over an attack graph. We presented a novel game formulation

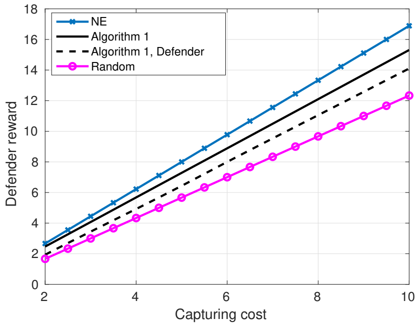

Fig. 4. Algorithm is yielding a comparable reward to the full game Nash equilibrium

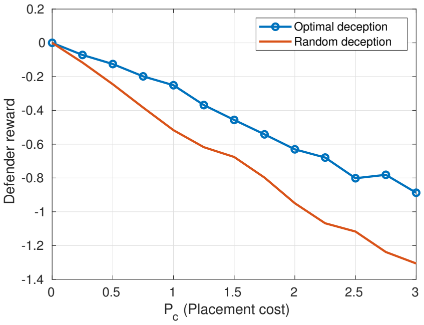

Fig. 5. Comparing defender reward at NE and random allocation at different placement cost

that captured the node values. In this paper, we investigated the trade-off between security cost and deception reward for the defender. The attacker decides which node to attack to maximize his reward while remaining stealthy. Nash equilibrium defense strategies are analytically characterized. The complexity of the general game is discussed and a scalable algorithm has been proposed to ensure that the defense approach is applicable to large scale networks. In our ongoing and future research, we will analyze the performance of the decomposition algorithm, investigate the dynamic version of the game over the graph with complete, one-sided and two-sided partial observation for both players.

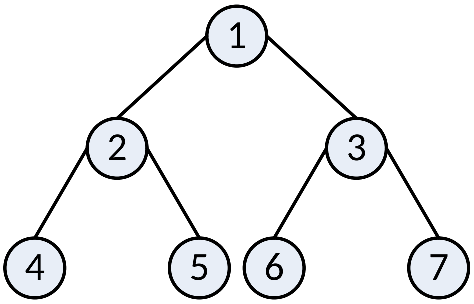

Fig. 6. A 7-node tree network topology

Acknowledgment

Research was sponsored by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-19-2-0150. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

References

[1] C. V. N. Index, “Global mobile data traffic forecast update, 2016–2021 white paper,” *Cisco: San Jose, CA, USA*, 2017.

[2] A. Kott, A. Swami, and B. J. West, “The internet of battle things,” *Computer*, vol. 49, no. 12, pp. 70–75, 2016.

[3] C. A. Kamhoua, “Game theoretic modeling of cyber deception in the internet of battlefield things,” in *2018 56th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 862–862, IEEE, 2018.

[4] N. C. Rowe and H. C. Goh, “Thwarting cyber-attack reconnaissance with inconsistency and deception,” in *Information Assurance and Security Workshop, 2007. IAW’07. IEEE SMC*, pp. 151–158, IEEE, 2007.

[5] P. Engebretson, *The basics of hacking and penetration testing: ethical hacking and penetration testing made easy*. Elsevier, 2013.

[6] G. F. Lyon, *Nmap network scanning: The official Nmap project guide to network discovery and security scanning*. Insecure, 2009.

[7] T. E. Carroll and D. Grosu, “A game theoretic investigation of deception in network security,” *Security and Communication Networks*, vol. 4, no. 10, pp. 1162–1172, 2011.

[8] Y. Li, Y. Xiao, Y. Li, and J. Wu, “Which targets to protect in critical infrastructures-a game-theoretic solution from a network science perspective,” *IEEE Access*, vol. 6, pp. 56214–56221, 2018.

[9] W. Xu, W. Trappe, Y. Zhang, and T. Wood, “The feasibility of launching and detecting jamming attacks in wireless networks,” in *Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*, pp. 46–57, ACM, 2005.

[10] A. Clark, Q. Zhu, R. Poovendran, and T. Başar, “Deceptive routing in relay networks,” in *International Conference on Decision and Game Theory for Security*, pp. 171–185, Springer, 2012.

[11] S. Jajodia, P. Shakarian, V. Subrahmanian, V. Swarup, and C. Wang, *Cyber warfare: building the scientific foundation*, vol. 56. Springer, 2015.

[12] H. Çeker, J. Zhuang, S. Upadhyaya, Q. D. La, and B.-H. Soong, “Deception-based game theoretical approach to mitigate dos attacks,” in *International Conference on Decision and Game Theory for Security*, pp. 18–38, Springer, 2016.

[13] M. Bilinski, R. Gabrys, and J. Mauger, “Optimal placement of honeypots for network defense,” in *International Conference on Decision and Game Theory for Security*, pp. 115–126, Springer, 2018.

[14] T. Zhang and Q. Zhu, “Hypothesis testing game for cyber deception,” in *International Conference on Decision and Game Theory for Security*, pp. 540–555, Springer, 2018.

[15] A. Schlenker, O. Thakoor, H. Xu, F. Fang, M. Tambe, L. Tran-Thanh, P. Vayanos, and Y. Vorobeychik, “Deceiving cyber adversaries: A game theoretic approach,” in *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, pp. 892–900, 2018.

[16] J. Letchford and Y. Vorobeychik, “Optimal interdiction of attack plans,” in *Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems*, pp. 199–206, International Foundation for Autonomous Agents and Multiagent Systems, 2013.

[17] D. O. T. ARMY, “Army support to military deception,” 2019.