APPLICATIONS OF GAME THEORY AND MULTI-AGENT REINFORCEMENT LEARNING IN CYBER SECURITY THREATS

R. ROBERT

Assistant Professor,

Dept. of Electronics and Communication Engineering,

Annai Velankanni College of Engineering,

Kanyakumari .

Email id: rrobertraj@gmail.com

Dr. V.V. VINOTH., M.E, Ph.D

Associate Professor,

Dept. of Electronics and Communication Engineering,

Annai Velankanni College of Engineering,

Kanyakumari.

Email id: vinfo.vv@gmail.com

Abstract- Impact of cyber-attacks is continuing to grow, then organizations to protect their digital data and the information they circulate or manage. Because of its, game theory and multi agent reinforcement learning has evolved for analyzing and modifying existing cyber protection methods to occur the best possible solutions. This paper addressed the new cyber security threats and primary target for cyber-attacks with optimal selection problem of attacker and sensor in cyber physical systems. Game theory analyses the model behaviors and study how attackers and defenders make decisions in a competing field.

Keywords- Cyber security, Game theory, Cyber physical system.

I. INTRODUCTION

In cyber security, game theory [21] is used to create many solutions and optimize them to a robust and long-term security environment at the organization level [2–4]. Using game theory, network of and reduce the risk to their valuable assets. Specifically based on game theory, it is possible to predict the attackers' strategy using intelligent models and to improve cyber security and the development of new intelligent systems.In Cyber-Physical Systems (CPS) Supervisory Control and Data Acquisition (SCADA) deployed for smart cities to monitor and control industrial processes. Industrial Control Systems (ICS) use advanced computing technology, sensors, control systems, and communication networks. with the technological industrial advances, ICS were protected and secured by isolation from the Internet and these systems became more distributed and exposed.

. II. LITERATURE REVIEW

Schlenker et al. [8] explored the critical issue of the cyber-alert allocation game to investigate this issue and demonstrated how to compute the defender's best options. If the attack occurs then alerts will be generated to the system. The system has a cyber allocation game (CAG) model for the cyber network protection domain, an NP-hardness proof for computing the optimal strategy for the defender, techniques to find the optimal allocation of experts to alerts in CAG in the general case and key special cases, and heuristics to achieve significant scale-up in CAGs with minimal loss in solution quality.

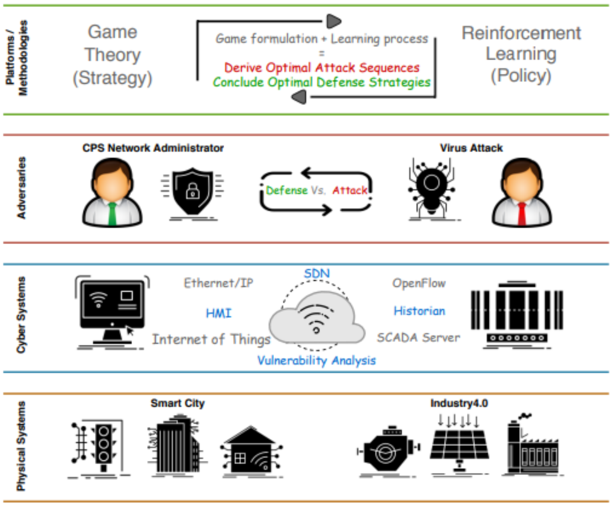
Alpcan and Basar [10], security and game theoretic approaches use quantitative models for making resource allocation decisions that balance available capabilities. It provides a mathematical foundation for making security risks in a principled manner. This game theory is applied in variety of systems such as water, electricity and communication networks.

Hemberg et al. [11] presented adversarially-hardened cyber defenses can be investigated using the dynamics of these cyber engagements. This can be competing for a coevolutionary mechanism in security contexts of network cyber security. This system is capable of pro-active cyber security against dynamic automated attackers.

.

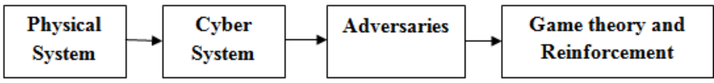
III. METHODOLOGY

The interconnection between these systems and organizational enterprise networks affected to cyber-attacks. For risk modeling and assessment many research focuses on new methodologies. Specifically hybrid game theory and reinforcement learning, control theory, and network optimization.



Many cyber-attacks follow a pattern built on repeating tactics or procedures over time. Infrastructure vulnerability control strategies, for example, can compete with current defensive systems and change their applications over time. In this sense, attackers and defenders engage regularly, and these interactions may be depicted using repetitive games, which are a type of dynamic game.

At the bottom, the physical system including smart cities, industry 4.0, Critical Infrastructures (CI), and IoT. Second, the monitoring and controlling the physical systems is done by the cyber systems layer which handles the operations related to the application of CPS and SCADA systems. The third layer is the CPS network administrator to reduce potential threats, and the rogue attackers trying to breach the cyber systems to cause physical damage. On top comes the methodology modelling layer.



II. BACKGROUND INFORMATION

A. Game Theory

Game Theory is an effective formal tool for strategic behavior analysis. Game theory describes agents that are selfinterested and interacting through a shared environment. Each agent has its description of the states of the world and which states it likes most. Each agent acts based on this description and has a utility function. The utility function quantifies the degree of preference across alternative actions given the other agents choices. An agent has decision-theoretic rationality in the sense that it acts to maximize its expected utility in the long run. The goal of using game theory is to reach a Nash Equilibrium state (NE). NE is defined as a state where no agent would like to deviate from without losing utility to other agents.

B. Multi-Agent Reinforcement Learning (MARL) MARL [6] is considered as an adaptation of game theory with the additional feature of machine learning. Learning in MARL is achieved through systematic trial and error in a shared and dynamic environment. The agents learn to choose actions that tend to increase their overall expected reward. A policy is a function that maps a state to an action. Optimal policies can be learned using a wide variety of algorithms including deep reinforcement learning. The goal of MARL is to derive optimal policies. MARL assumes that the game converges to a Nash Equilibrium after the learning phase. brid approach is divided into two levels: first, a higher game strategic level and second, a lower battlefield level.

C.Strategic level

We model the strategic level using an imperfect information extensive form game. The states of this game are the overall security status: Low, medium, high, and critical danger. Obviously, the defender does not know for sure what is the actual state (e.g., the attacker has discovered a zero-day vulnerability). The attacker does not know what the state is since he does not have the full information about the target network or all the defender countermeasures. Therefore, we consider an imperfect information model. This game has an extended form since the defender will try to recover from corrupt states to the original (low) state. The attacker will try to reach the critical state. Therefore the game has multiple stages. The defender chooses the strategy in terms of countermeasures to impede the progress of the attacker. The strategy is translated into a set of actions to choose from at the Battlefield level. The attacker selects the strategy in terms of attack methodologies. Attacker strategies are translated as a set of actions, vulnerabilities, and penetration tools at the Battlefield level.–

B. Battlefield level

This level is modeled using MARL and represents a multistage stochastic game with a learning component; each agent has a game turn with a random transition probability and creates a new scenario for the other agent to act accordingly. The state of the game is composed of the state of the defender based on the alerts, and the state of the attacker based on its attack tree. The nature of this game is stochastic, since some attacks can fail with some probability, and some countermeasures can fail with another probability.

1) Network Architecture: The game model is based on a modern CPS network with an architecture segregated into four subnets and divided into two layers describing a real CPS network. First, a cloud-based network hosting the web server, email server, and the database server of the CPS organization. Second, an enterprise network that hosts all the office’s computers for the CPS enterprise. Third, a SCADA network that hosts the SCADA Server responsible for data acquisition, the Human Machine Interface (HMI) used by operators to monitor sensor values and take commands accordingly, and the SCADA historian where acquired data are stored. Finally, The field network contains the goal of the attacker, which is the control units represented by a Programmable Logic Controller (PLC). PLCs execute commands sent from the SCADA network on the physical processes and also retrieves sensor values from sensors deployed all over the industrial process line to be afterward sent to the SCADA network for processing and storing. Each host deployed on the different subnets of the CPS network is explicitly assigned a unique private directory acting as its storage unit. These directories will be used as a ground for virus spreading among the hosts, the four subnets, and the two layers of the CPS network. To correctly mimic a modern CPS network with its two different layers and its subnets, MiniCps [12] is used as a network simulator to target network communication, control logic, and physical layer interaction. Using MiniCps features, a terminal can be launched at each host to execute a specific network command. In this paperwork, we focus on implementing a virus spreading among the hosts; Netcat command was used to read and write files using a TCP connection, which allows the virus to pass from a directory to another until reaching its final destination. 2) Vulnerability Analysis: The CPS network that is designed and implemented in this paperwork is a vulnerable CPS network segregated into four subnets. The cloud-based subnet is hosted on the cloud by an outside organization such as Amazon Web Services (AWS) [13], which is a modern practice adopted by all big firms these days. Also, the rest of the subnets are hosted locally in the CPS organization. The base metric provided by Common Vulnerability Scoring System (CVSS) [14] is adopted to quantify vulnerabilities on network nodes in this paperwork. More specifically, the exploitability score that describes best the complexity of exploiting a vulnerability.

Game Formulation:

ATTACK MODEL

In modeling an attack, we are considering parties with a conflict of interests: the attacker and the defender. The defender, often a system administrator, manages the system. The main interest of the defender is to secure the cyber infrastructure from malicious activities. The attacker, on the other hand, is a malicious opponent who attempts to compromise the target system. We model the interaction between the attacker and the defender based on data on actual security incidents..

Attacker

The attacker is an opponent who accesses the system with the intention of threatening its security. Attacks can vary from a single action to a sequence of activities. In this paper, we limit our interest to attacks that consist of multiple activities that lead to an ultimate goal. Attack State ASx represents the state of the attack, i.e., the depth/degree of intrusion. Each attack state is assigned a numeric value(reward) which quantifies the damage to the target system. The bigger the impact, the more severe the damage to the system and/or the greater the unauthorized control over the system. Transition from one state to another depends on the result of the action. Activity A is a set of actions ai available to the attacker. It can lead to malicious control over the system, or if the attacker decides to remain in the current state, the transition will result in a loop. The set of available activities in state ASx is denoted by Ax. Therefore, Ax is a subset of A. The causal relation between activities and attack states can be represented as a state diagram. Transition Matrix Pa(s, s0) is the probability that an action from state s will lead to a transition to the next state s'. In an attack model, a transition matrix represents the probability of a successful attack. Depending on the monitoring system configured on the defender's side, an attack can be either detected or missed. The transaction matrix models the uncertainty of the result of an action. Immediate Reward Ra(s, s0) is the reward of the attacker as a result of a transition from state s to s' for performing action a. The reward is a quantitative representation of the earnings that the attacker can get from a successful attack.

B. Defender

The defender is a party that is in charge of making proper responses to secure the system from malicious attacks. The defender has a set of monitors to protect the system. The main objective of this player is to make prooper responses in a preemptive manner based on a limited view of the system status, relying on monitors. Attack State DSx represents the state of the attack from the defender's perspective. The observations that defenders use rely on the monitoring systems, and lack the granularity needed to reveal the details of users' actions. Defender Action D is a set of actions(d) available to the defender in a given state. For security incident detection and response, a monitor detects changes in system status. However, such detections do not directly map to the attacker's definite actions. The monitor may miss an action (false negative) or misidentify a benign action as malicious (false positive). Hence, the defender needs to take an appropriate action while relying on imperfect information. Assuming that there are proper responses for each action, we abstract the defender action to either "Reponse" or "No Response," where "No Response" is useful for monitored events that are hard to differentiate from benign ones, and/or events that do not cause immediate harm to the system.

C. Attacker-Defender interaction

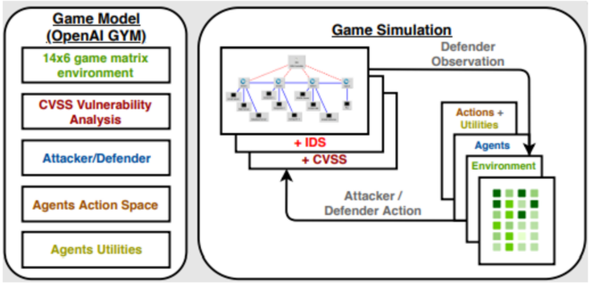
While each attacker has a logic flow for making decisions, his or her decisions are not independent, but are related to the opponents decision process. Hence, we model the interaction between the two players. In Figure 1c, we show a subset of the security game. Once an attacker has taken an action, the defender chooses his or her action based on the information from the monitoring system. An attackers action results in a transition to the intended state only if the defender does not make a proper response. Once the defender has responded to the observed action, the attacker is forced to transit to the default state. Assuming a zero-sum game, a successful attack will result in an immediate reward, and the defender will have a symmetric loss. As a result of the execution of the attack, the attack state will change accordingly. Otherwise, if the defender detects the attack and makes a proper response, the attack state will be reset to the default for the identified attacker. In that case, a reward will be assigned to the defender, with an equivalent loss to the attacker.

Next is a six-tuples defining the essential parameters of the proposed game model: the adversaries agents tuple, a set of states representing the 14x6 game model matrix (14 nodes, five vulnerabilities, level of protection), two sets for attacker and defender action samples, a set of reward corresponding to each state, and finally a set of transition probabilities that is represented by the exploitability score of the vulnerabilities.

$$\begin{cases} \textit{Agents:} & X = \{X_a, X_d\} \\ \textit{States:} & S = \{\{S_1^1, ..S_1^6\}, \ldots, \{S_N^1, ..S_N^6\}\} \\ \textit{Att. actions:} & A = \{\{A_{S_1^1}, ..A_{S_1^5}\}, ..., \{A_{S_N^1}, ..A_{S_N^5}\}\} \\ \textit{Def. actions:} & D = \{\{D_{S_1^1}, ..D_{S_1^6}\}, ..., \{D_{S_N^1}, ..D_{S_N^6}\}\} \\ \textit{Rewards:} & R = \{\{R_{S_1^1}, ..R_{S_1^6}\}, ..., \{R_{S_N^1}, ..R_{S_N^6}\}\} \\ \textit{Trans. prob:} & P = \{\{P_{S_1^1}, ..P_{S_1^5}\}, ..., \{P_{S_N^1}, ..P_{S_N^5}\}\} \end{cases}$$

The defender uses extra parameters during the simulation, which helps determine a win-game for the defender; a value W that returns zero if the virus is not detected and one if identified by the IDS, and another value T that indicates if the game is in a terminated state. In addition to value Y that delimits the steps of an attacker, and a success rate of V for the attacker actions. N is used to indicate the index of the last node on the system:

$$\begin{cases} \textit{IDS:} & W \in \{0 : \textit{non-detected}, 1 : \textit{detected}\} \\ \textit{Terminal state:} & T \in \{0 : \textit{non-terminal}, 1 : \textit{terminal}\} \\ \textit{Maximum steps:} & Y \in [0; 99] \\ \textit{Success rate:} & V \in [0; 99] \\ \textit{Last node index:} & N = 14 \end{cases}$$



simulator. Fig. 3: Proposed framework architecture using OpenAI Gym toolkit, and MiniCps network

CPS network that is the closest in architecture and features to modern CPS networks and, at the same time, simple to be used for simulation and learning purposes. Fig. 3 shows the proposed framework architecture that is created with the combination of MiniCps network simulator, and OpenAI Gym toolkit to achieve the requirements of the proposed hybrid approach. First, the CPS network is created using MiniEdit, a helper tool in MiniCps [12] that allows the creation of a network with a drag and drop functionality and at the end generates a Python file describing the designed network with all its hosts, switches, and links. Each network device from Fig. 2 was implemented as a host connected by switches for routing. Upon generation, the file is edited to support explicitly (1) private directories for each hoston the CPS network,

(2) IDS implementation to detect attacks and return proper observation for learning purposes, and

(3) integration of vulnerability analysis on the hosts and switches of the CPS network.

Using OpenAI Gym, the environment of the proposed game model is created; it is a 14x6 matrix representing the 13 nodes of the network plus a starting node as rows and the set of five vulnerabilities plus the IDS strength for each host on the network as columns, at each node the first five cells of the matrix contains the exploitability score of a vulnerability on a specific node and the last cell contains the strength of the IDS on each node. These exploitability scores represent the defense values of the defender and how the network is immune against attacks. Also, using the Gym library, the action spaces for both agents are determined, the rewards for each node of the environment are set, and helper functions are created to apply the chosen action on the designed CPS network that is fed into the game model. By merging the CPS network created using MiniCps and its add-ons (Directory creation, IDS integration, and vulnerability analysis) with the created game model using OpenAI Gym, a robust framework is designed to conduct simulation, execute a different kind of strategies, and apply learning algorithms to derive optimal defense policies. In this study, Q-learning was used to achieve the learning element. The environment design has a limited number of states with a limited number of actions allowed for agents; this argues the use of an algorithm based on tabular representations instead of an algorithm based on neural networks. In addition to the problem that neural networks have with adversarial learning, where the network is slow in adapting to the continually changing environment [9].

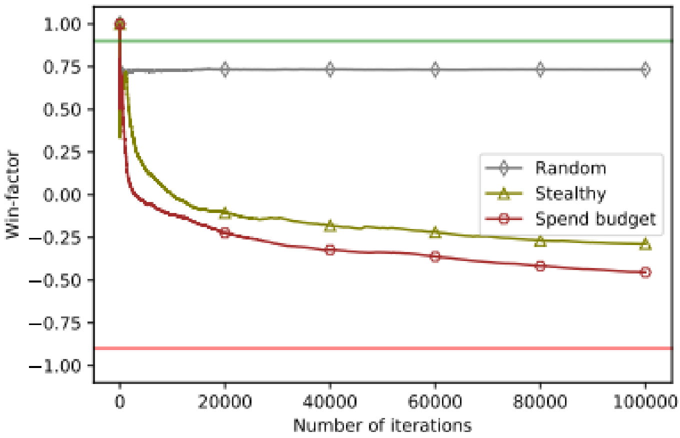


Fig. 5: Win-factor for learned attacker vs. random defender

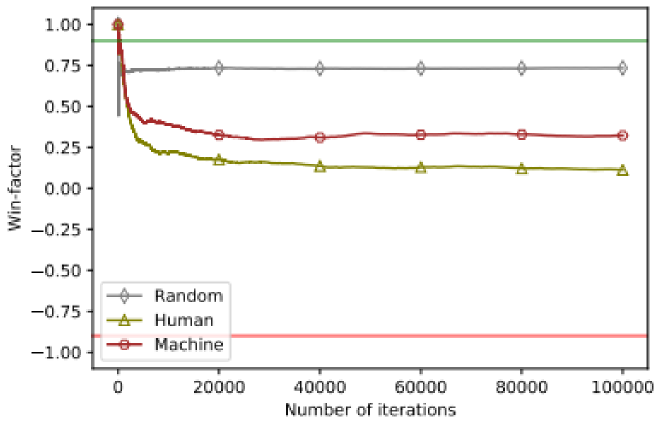


Fig. 6: Win-factor for learned attacker vs. learned defender

V.CONCLUSION

Future malware will undoubtedly be equipped with automated learning components. This paperwork takes this assumption and proposes a framework based on a hybrid approach using game theory and RL to model an adversarial game for cross-layer virus spreading in CPS networks. The work presented makes use of previous literature and adds what we consider more realistic features such as the simulation of a virus spreading, vulnerability analysis, zero-day exploits, and a larger CPS network architecture. The formulation includes a network architecture with different layers and subnets that mimics a real and modern CPS network. The utilities of the game are the cumulative rewards that can be achieved by the RL agents after the learning phas.

MARL was applied using Q-learning for both agents to learn optimal policies. The obtained results show the ability of the defender to derive optimal defense policies based on a human-selected strategy to reduce losses and prevent viruses from spreading across the CPS networks. A mixed defense strategy can lead the game to a NE, where the attacker would not like to change behavior since it can be countered more easily. A particular focus in this paper was put on the RL level in terms of design and simulations. In future work, we will focus more on the strategic level, by dissecting the extended form security game with imperfect information.

REFERENCES

1. Grønbæk L., Lindroos M., Munro G., Pintassilgo P. Basic concepts in game theory. In: Grønbæk L., Lindroos M., Munro G., Pintassilgo P., editors. Game Theory and Fisheries Management Game Theory and Fisheries Management: Theory and Applications . Berlin, Germany: Springer International Publishing; 2020. pp. 19–30. [CrossRef] [Google Scholar]
2. Akinwumi D. A., Iwasokun G. B., Alese B. K., Oluwadare S. A. A review of game theory approach to cyber security risk management. Nigerian Journal of Technology . 2018;36(4):p. 1271. doi: 10.4314/njt.v36i4.38. [CrossRef] [Google Scholar]
3. Alpcan T., Vorobeychik Y., Baras J. S., Dán G. Decision and game theory for security. Proceedings of the 10th international conference, GameSec 2019; November 2019; Stockholm, Sweden. [CrossRef] [Google Scholar]
4. Do C. T., Tran N. H., Hong C., et al. Game theory for cyber security and privacy. ACM Computing Surveys . 2018;50(2):1–37. doi: 10.1145/3057268.
Schlenker A., Xu H., Guirguis M., et al. Don't bury your head in warnings: a game-theoretic approach for intelligent allocation of cyber-security alerts. Proceedings of the 26th International Joint Conference on Artificial Intelligence; August 2017; Melbourne, Australia. pp. 381–387.
Alpcan T., Vorobeychik Y., Baras J. S., Dán G. Decision and game theory for security. Proceedings of the 10th international conference, GameSec 2019; November 2019; Stockholm, Sweden.
Hemberg E., Zhang L., O'Reilly U.-M. Exploring a adversarial artificial intelligence for autonomous adaptive cyber defense. In: Jajodia S., Cybenko G., Subrahmanian V. S., Swarup V., Wang C., Wellman M., editors. *Adaptive Autonomous Secure Cyber Systems* . Springer International Publishing; 2020. pp. 41–61.