SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ DERGİSİ

Sakarya University Journal of Science
SAUJS

ISSN 1301-4048 e-ISSN 2147-835X Period Bimonthly Founded 1997 Publisher Sakarya University
http://www.saujs.sakarya.edu.tr/

Title: Reinforcement Learning Applications in Cyber Security: A Review

Authors: Emine CENGİZ, Murat GÖK

Recieved: 2023-01-17 00:00:00

Accepted: 2023-02-12 00:00:00

Article Type: A Review Article

Volume: 27
Issue: 2
Month: April
Year: 2023
Pages: 481-503

How to cite
Emine CENGİZ, Murat GÖK; (2023), Reinforcement Learning Applications in Cyber Security: A Review. Sakarya University Journal of Science, 27(2), 481-503, DOI: 10.16984/saufenbilder.1237742
Access link
https://dergipark.org.tr/en/pub/saufenbilder/issue/76551/1237742

Sakarya University Journal of Science 27(2), 481-503, 2023

Reinforcement Learning Applications in Cyber Security: A Review

Emine CENGİZ *[1], Murat GÖK[1]

Abstract

In the modern age we live in, the internet has become an essential part of our daily life. A significant portion of our personal data is stored online and organizations run their business online. In addition, with the development of the internet, many devices such as autonomous systems, investment portfolio tools and entertainment tools in our homes and workplaces have become or are becoming intelligent. In parallel with this development, cyberattacks aimed at damaging smart systems are increasing day by day. As cyberattack methods become more sophisticated, the damage done by attackers is increasing exponentially. Traditional computer algorithms may be insufficient against these attacks in the virtual world. Therefore, artificial intelligence-based methods are needed. Reinforcement Learning (RL), a machine learning method, is used in the field of cyber security. Although RL for cyber security is a new topic in the literature, studies are carried out to predict, prevent and stop attacks. In this study; we reviewed the literature on RL's penetration testing, intrusion detection systems (IDS) and cyberattacks in cyber security.

Keywords: Cyber security, reinforcement learning, penetration testing, IDS, cyberattack.

1. INTRODUCTION

Cybersecurity can be defined as technologies and processes that help protecting the integrity, confidentiality and availability of networks and data in computer systems against cyberattacks or unauthorized access [1]. Cyber security has become one of the most important problems in cyberspace [2]. Recent developments in information technologies, communication networks, the internet of things, cloud technology, increase in mobile internet and development of hardware of devices have revealed security vulnerabilities and uncertainties. This situation causes systems not to function, economic damage and danger to cyber security.

In favour of maximising the level of security of system assets, it is required to build up innovative and intelligent defense methods that are able to overcome cyber threats [3]. For this, it is necessary to obtain the historical and current security status data of the system and make intelligent decisions that provide

* Corresponding author: emine.cengiz@yalova.edu.tr (E.CENGİZ)
[1] Yalova University
E-mail: murat.gok@yalova.edu.tr
ORCID: https://orcid.org/0000-0002-6695-9500, https://orcid.org/0000-0003-2261-9288

security management and control. Machine learning (ML) is a method applied to both attack and defense. This method is used to make the defense mechanism smarter, more durable and more efficient [4]. At the offensive level, however, it complicates the attacks to get through the defensive methods easily.

It is known that simple algorithms are not enough for cyber security software to fulfill their task. Many studies conducted on the Reinforcement Learning (RL) method, which leads today's algorithms, show that its importance in cyber security is increasing. RL is a purposeful ML approach that learns what to do. The RL agent directly relates to the environment to reach a set target, imitating the human learning process [5]. The agent learns by trial and error and uses experience to improve its behavior [6]. RL has been used in numerous disciplines such as robotics [7-9], control systems [10], advertising [11], video games [12-14], autonomous vehicles [15,16], autonomous surgeries [17, 18].

The main goal of this research is to compose a narrative review of studies, which provide an overview of what is known about a particular topic and are often topic based [19], in the field of cyber security using RL. We used selective search method which surveys only that literature and evidence that are readily available to the researchers [20].

The rest of this paper is organized as follows. In Chapter 2, general information about the RL algorithm is given. In Chapter 3, RL applications in cyber security are discussed in detail. The conclusion of the article is summarized in Chapter 4.

2. REINFORCEMENT LEARNING

RL provides a suitable study for modeling complex control problems and solving these models using learning algorithms [21]. Unlike other methods of ML, RL is a reward-based learning method that interacts with the environment [22]. The learning machine, called the agent in RL, reacts to the situations it encounters. In consequence of this response, it receives a numerical reward value. Figure 1 demonstrates the basic structure of the RL.

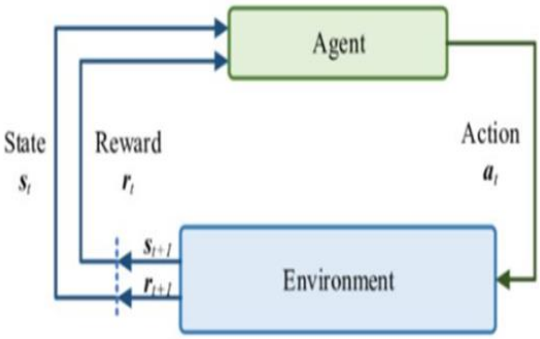

Figure 1 Reinforcement learning

Table 1 RL parameters

Parameters	Definition
s_t	state at time t
$a_t \in A$	a action taken from action space A
$s'_t = s_{t+1}$	new state passed with action a at time t
$r_t = r(s_t, a_t, s'_t)$	in case s_t, the reward obtained by switching to the s'_t state with a_t decision
$\pi(s,a) = P(a_t = a \mid s_t = s)$	probability of the agent making decision a_t in case s_t

The agent, that has no information about the environment, makes a choice of action in accordance with the situation it is in. This choice is evaluated by the environment and the agent moves into a new state. The agent evaluates the reward or punishment it received

from the action it made in the previous situation by its own decision-making mechanism and produces a new action for the new situation. This cycle continues until the agent completes the learning process. The mathematical symbols that are used in the learning process are presented in Table 1.

The goal of RL is to learn which action to take in any given situation. In order to find out this, it is necessary to calculate the quality of movement a. In case s, the quality of action a is defined as the total reward expected when acting within the framework of the decision-making function π. R_t is the summation of all rewards available from time t. The sum of these rewards is shown in Equation 1.

$$R_t = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \cdots = \sum_{k=0}^{\infty} \gamma^k r_{t+1+k} \quad (1)$$

The discount rate γ is used when calculating the reward. The discount rate determines the importance of future reward value in decision making and takes a value between $0 \leq \gamma < 1$. In the case of $\gamma=0$, the agent makes a decision considering the highest reward value at that moment, while in the case of $\gamma>0$, it chooses its actions taking the future rewards into account. The determination of this rate directly affects the learning of the agent.

RL algorithms use an estimation of value functions that shows how important the states are to the agent. Value functions are calculated through systems called policies, which probabilistically determine which actions to choose.

Status value and action value functions are calculated from the status and rewards obtained by the steps taken in line with the policy. The state value function is shown in Equation 2. While the state value function is in state s, it returns the expected value of state s when policy π is fulfilled.

$$V^{\pi}(s) = E[\sum_{k=0}^{\infty} \gamma^k r_{t+1+k} \,|s_t = s] \quad (2)$$

When the action value function is in the state of s, it returns the expected value of the state-action pair when it chooses the action a using the policy of π. The action value function is shown in Equation 3.

$$Q^{\pi}(s,a) = E[\sum_{k=0}^{\infty} \gamma^k r_{t+1+k} \,|s_t = s\,, a_t = a] \quad (3)$$

For policy π and state s values, Equation 3 shows the consistency condition among the value of state s and the value of states to be reached from s. This gives Equation 4, the Bellman equation.

$$V^{\pi}(s) = \sum_a \pi(a|s) \sum_{s',r} p(s', r|s, a)[r + \gamma V^{\pi}(s')] \quad (4)$$

This equation simplifies the calculation of the value function. Because we can find the best solution to a complex problem by dividing it into simpler and iterative sub-problems instead of summing multiple time steps [20].

Deep Reinforcement Learning (DRL)

The integration of deep learning and RL is a breakthrough that Google DeepMind initiated and spearheaded to form an intelligent agent capable of defeating a professional human player in [23] 49 Atari games. DRL is a revolutional technique in RL that can solve complex computational task [24]. The DRL model is given in Figure 2.

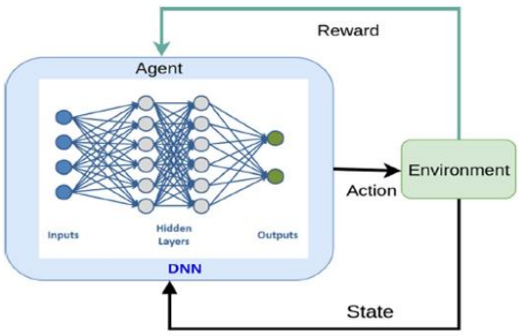

Figure 2 Deep reinforcement learning

The learning system is the same as the RL methods. However, some parts of the system are modeled by deep learning technique. For instance, deep learning is suitable to be used to acquire the amount of reward corresponding to a given state-action pair. In addition, deep learning can increase the intelligence of RL agents and accelerate the agent's ability to optimize policy [25]. DRL has been used in control [26], resource management [27], robotics [28, 29] and many other applications.

3. RL IN CYBER SECURITY

Increasing levels of interaction between cloud computing and machines have resulted in a remarkable increment in the number and complexity of cyberattack cases. Therefore, securing user data, privacy and devices have become an important issue nowadays. Numerous RL models have been presented in the literature for various applications of cybersecurity. This section provides a comprehensive review of RL-powered solutions for penetration testing, intrusion detection systems, and cyberattacks.

3.1 RL in Penetration Testing

Penetration testing (PT) are safety tests performed by "authorized" persons and “legal” entities in order to detect logic errors and vulnerabilities of digital assets (network, website, application, database) to prevent exploitation of security vulnerabilities by malicious people and to make systems more secure [30]. Checking and reporting security vulnerabilities in information systems by a third eye plays an important role in ensuring security. The steps that take place when PT is applied are shown in Figure 3.

Although systems evaluated using PT vary, the same general steps are followed in all cases. When a successful attack is carried out, a specific set of attack actions is reported. With this report, security vulnerabilities can be minimized or completely avoided by system administrators and developers.

It is possible to automate the discovery, exploitation and identification steps by imitating experts in PT. The need to develop autonomous pentest solutions has become important to ensure that the pentest results implement extensive testing of attack surfaces. Autonomous pentesting is an emerging research area. An important point here is the method to create the attacks. Sarraute et al. [31] simulated the attack planning problem with regard to Partially Observable Markov Decision Processes (POMDP). POMDP allows to evaluate available information and to use scan actions intelligently as a part of the attack. It is recognised that POMDPs are not scalable for many network nodes. In this study, the PT process is divided into four levels using the multi-level architecture called 4AL: individual machines, attacking components, decomposing the network and subnetworks. In this study, the scalability problem of POMDPs has been tried to reduce by separating the network into double link components consisting of more than one subnet. Sarraute et al. [32] designed POMDP in the pentest problem in another study. POMDP has been used to overcome the missing information limitation by creating attack plans when missing information and uncertainties are given to the planner. It has also been observed again that the POMDP-based solution does not scale well for large networks. POMDPs are complex and require large computational resources. As a solution, Hoffmann [33] presented a common platform between POMDPs and classical planning called Markov decision processes (MDP) in his study. Unlike previous studies, actions do not scan. Each action result is assigned by a probability value that is independent from the

estimates of the host configuration, and this value refers to the level of uncertainty in which the attacker initiated the action. It has been observed that Planning Domain Definition Language (PDDL) is not endowed to overcome these uncertainties. For this, he proposed a language which is like PDDL that enables probability values within actions. Schwartz and Kurniawati [34] showed that RL can search to find the cause of the network exploit and the attack policy on all target machines. They used a Network Attack Simulator (NAS) and three different Q-learning algorithms in their study. These are tabular Q-learning using Upper Confidence Bound (UCB) action selection, tabular Q-learning using ε-greedy action selection (tabular ε-greedy) and deep Q-learning (DQL) using a single layer neural network and ε-greedy action selection. The network created with a NAS consists of elements including connections, subnets, services, hosts and firewalls.These constituents enable the network simulator to run on different systems. Algorithm performance started to decline when there were more than 43 machines in the network. As a result of the study, it was observed that tabular RL algorithms do not scale well in large networks with many machines, whereas DQN does not scale well when the number of actions increases.

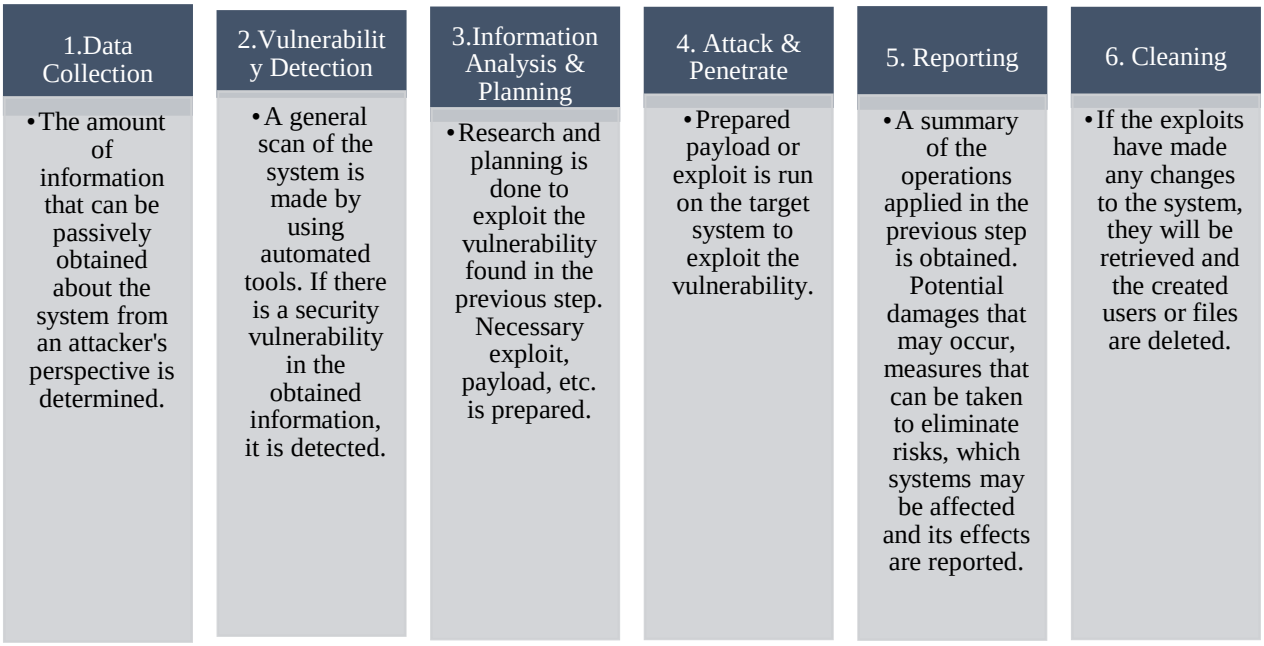

Figure 3 Main steps of penetration testing

Ghanem and Chen [35] recommend an "Intelligent Automated Penetration Testing System (IAPTS)" that combines with industrial PT frameworks to enhance the performance and accuracy of medium and large network infrastructures. IAPTS targets to save human resources while providing improved outcomes with regard to reliability, test frequency and time. As in previous studies ([30-31, 33]), an environment consisting of 10-100 machines was used in this study. They concluded that RL outperforms the -abilities of any PT specialist concerning attack vectors, time, reliability and accuracy of the outputs.

The complexity and uncertainty of penetration testing can be determined using the RL environment [34]. Previous studies [31-32, 34-35] using the RL method have been

observed to be mostly successful in small networks. Chowdary et al. [36] designed a model to learn efficient pentesting schemes in large networks. They worked on an autonomous security analysis framework which assists in reducing the manual work done on PT. Penetration testing framework (ASAP) was used to generate the attack graph. ASAP creates autonomous attack plans and reveals undetected stealth attack paths in a manual test. In their study, one of the RL algorithms, Deep-Q Network (DQN), was also used to determine the most appropriate policy in PT. It has been observed that the created framework is more scalable on a large network compared to existing studies. Nguyen et al. [37] propose a double-agent architecture (DAA) technique to escalate the size and performance of the network. DAA uses the MDP model to attack environments. The purpose of DAA is to implement PT in entensive network systems with about 1000 machines. It is observed that the ability of DAA to attack hosts successfully is 70% if it reaches 1024 hosts and 100 services, and up to 81% in networks with less than 10 available services and 1024 hosts. The double agent structure presents in this work is shown in Figure 4.

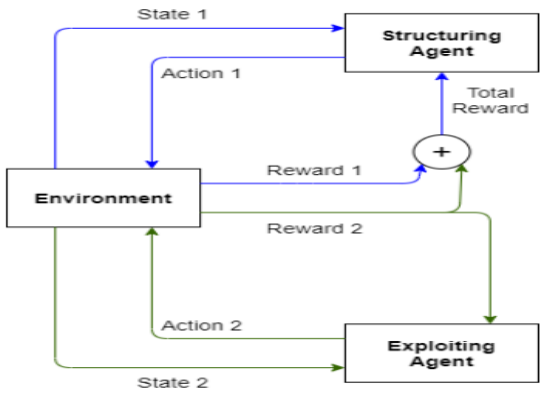

Figure 4 Double-agent structure [37]

Here, as a first step the structuring agent monitors the environment and receives a state1. If the agent considers that the state information is not sufficient, it continues with its structured discovery actions, including scanning of hosts and subnets. Meanwhile, the agent receives a reward1 value from the environment instantly and uses it to evaluate the accuracy of the action. Otherwise, if the agent concludes that it is allowable to gather information or make use of the services because of the lack of service information, it will not make direct decisions and will trigger the exploit agent. The exploiting agent uses the state of the selected host (state2) as the input. If the selected host considersthat the status information is not sufficient for the agent, hosts are continued to scan, otherwise it can take advantage of the host with the suitable service. After its action, the exploiting agent gets the reward 2 value from the environment to update its policy. The structuring agent also uses the sum of reward 1 and reward 2. If the structuring agent's chosen action is to scan hosts or subnets, the reward2 value is taken as 0. Finally, the structuring agent uses the sum of reward 1 and reward 2 to update its policy.

RL has demonstrated its capability to find the optimal way to attack. However, creating a correct model of exploitation and a realistic training simulator for agents are required [34]. Zennaro and Erdori [21] presented a PT approach that uses variable RL techniques in a simulation to obtain the hidden flag in the flag competition and overcome cybersecurity challenges. The primary goal of their research is to see the suitability of using different RL techniques for PT. Neal et al. [38] presented a groundwork for conducting PT steps against Microgrid (MG) control algorithms using RL. MGs are small-scale power systems with their own energy sources, outputs, and loads with certain limits. Within MGs, the digital infrastructure used to transfer data and execute control commands is compromised under a cyber-attack. In a simulated MG, the RL agent is trained to find malicious input that harms the MG controller.

Yang and Liu [39] formulate the Multi-Objective Reinforcement Learning (MORL) model in PT in their study. They used NAS and Cyber Autonomy Gym for Experimentation (CAGE) as the PT simulation platform. In PT, various types of attacks and agents with different behaviors are produced using Chebyshev criticism. A model is presented that increases agents' adaptability to future exploration and reduces their attention to previous actions. In this proposed model, it is shown that more information is collected from the network, thus enhancing the security level of the target network.

3.2 RL in Intrusion Detection Systems

With the universal use of computer and internet technology, the number of websites and web-based applications has increased rapidly. Sharing many important elements such as information, ideas and money through websites and applications provides convenience for people. However, material and nonmaterial losses may occur due to system and security vulnerabilities in applications. Many tools have been developed to prevent this situation and ensure safety. Security solutions were tried to be developed by using software such as firewall and virus programs, but it was not sufficient [40]. For this reason, it is aimed to analyse possible dangers with Intrusion Detection Systems (IDS) in addition to existing software. IDS are software products that are used to identify attacks by controlling the activities of a network or system, provide information about these attacks, and report attack attempts so that security analysts can analyse them better. The general form of the IDS [41] is given in Figure 5.

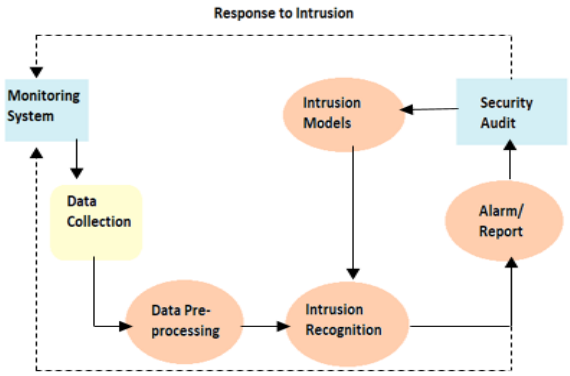

Figure 5 General structure of the IDS system

IDS are classified by two methods according to the installation and the detection mechanisms of the systems. According to the installation of the systems, it is divided into two as network- based IDS and host-based IDS. Network-based IDS is deployed at key points in a network to display the traffic between devices and computers on the network. Host-based IDS, on the other hand, works within a single computer and monitors the traffic coming from the system to the computer and its effects on the system. It is also divided into two, according to the detection mechanism of the system: signature-based IDS and anomaly-based IDS. Signature-based IDS store known attacks in pre-created signature databases and classify incoming samples by checking this database. In anomaly-based IDS, user profiles are created for the user group or for each user separately. A threshold value is determined for these profiles by means of various machine learning methods or mathematical models. If a transaction on the network deviates from this threshold significantly, it is considered an attack and an alarm is triggered.

With the lack of immediate response to dynamic intrusions and the development of attack methods, the RL method has started to be used in IDS. Xu and Xiu [40] proposed the RL method for Host-based IDS implementing the order of system calls. The Markov Reward Process (MRP) was used to model the

behaviour of system calls and the intrusion detection problem. A different learning algorithm that uses linear functions is applied to estimate the value function of the MRP. Xu [42] proposed a new sequential anomaly detection technique based upon temporal difference learning, in that multi-stage intrusion detection of cyber-attacks is considered as an implementation case. An MRP model was created for determining the anomalies and alarming process of the datasets. If the reward function is defined correctly, the anomaly possibilities of the datasets are equivalent to the value function of the MRP. This study was compared with different machine methods. It has been observed that the prediction accuracy can be enhanced even if the number of labelled training data is small.

Deokar and Hazarnis [43] conducted a study on the shortcomings of signature-based and anomaly-based detection methods. The authors proposed an IDS capable of recognizing attacks by combining the features of these two methods by using log files. In the proposed IDS that use the RL method, log correlation techniques and association rule learning are used together. RL is used to reward the system when it chooses log files that have anomalies or signs of attacks and it allows the system to select more suitable log files when looking for traces of attacks. Otoum et al. [44] propose a big data-oriented IDS approach in Wireless Sensor Networks (WSN) by utilizing the RL algorithm in a hybrid IDS framework. The research examines the efficiency of the RL-based IDS. Results are compared with Adaptively Supervised and Clustered Hybrid IDS (ASCH-IDS). Test results indicate that RL-IDS is able to reach 100% success in detection, accuracy and precision-recall rates.

Caminero et al. [45] worked on the first implementation of adversarial reinforcement learning to enable real-time prediction of attacks and detect intrusions. A new application that integrates the behaviour of the environment into the learning process of an improved RL method is provided. In aforementioned study, (NSL-KDD and AWID) datasets were used. The presented model was compared with different ML algorithms and it was observed that it outperformed other models on weighted accuracy (> 0.8) and F1 score (> 0.79) measures.

Sethi et al. [46] presented a context-adaptive IDS that maintains the balance between accuracy and false positive rate (FPR). This system has multiple independent RL agents deployed on the network to detect and classify complex and new attacks accurately. In the study, experiments were carried out using NSL-KDD, UNSW-NB15 and AWID dataset, showing higher accuracy and lower FPR compared to up-to-date solutions. In the model, the resilience of the system against attacks was analysed and only a slight drop in accuracy was observed compared to existing models.

Alavizadeh et al. [47] combined Deep Q-learning-based (DQL) RL with a feedforward neural network to detect and classify attacks. In the presented method, hyperparameters of a DQL agent like discount factor, batch size and the number of learning episodes are analysed to improve learning capabilities. In the study, using the NSL-KDD dataset, they obtained the best performance results in the case of 250 episodes of learning and a discount factor of 0.001. In their study, comparisons were made with other machine learning approaches to detecting different classes of intrusion, and they observed that their proposed model performed better (more than 90%).

Alawsi and Kurnaz [48] proposed a method grounded on measured Quality of Service (QoS) to evaluate the services given in the network to protect the network. QoS is evaluated periodically based on the services provided in the network and is used to calculate the reward value used to train the neural network. Decisions made for packages are reassessed based on their QoS value, as the goal is to increase the value of the collected reward. All decisions made by the neural network are updated when there is a decrease in the QoS value. CICIDS2017 dataset was used in the study. While the F1 score was 0.51 when classification-based neural network was used, an F1 score of 0.96 was obtained in this study.

3.3 RL in Cyber Attacks

Today, cyber-attacks and cyber security are among the most important digital transformation issues. The internet platform being so wide and easily accessible has created positive and negative effects. Cyber-attack is the name given to all of the attack actions made by one or more computers towards the opposite computers or networks, using various methods to steal, change or destroy data. It is possible to prevent these attacks or create defenses.

1) DoS and DDoS:

Denial of Services (DoS) and Distributed Denial of Services (DDoS) are types of attacks that are carried out against a target, hindering the system from functioning and preventing users from accessing the system. Attackers can send numerous requests to a database or website, keeping the system busy and causing systems to crash. DDoS, on the other hand, occurs when these attacks are made from more than one computer.

Xu et al. [49] used Hidden Markov Models (HMM) and RL to separate valid traffic from DDoS attacks based upon the source IP addresses being tracked. To detect DDoS attacks earlier, detection agents are located at network nodes or near DDoS attack sources. HMMs are used to generate regular traffic grounded on the frequencies of new IP addresses. The RL method is proposed to calculate the optimized information exchange between multiple distributed detectors.

Malialis and Kudenko [50] used the multi-agent router throttling method based on the SARSA algorithm for DDoS attacks in their study. They introduced this method by teaching multiple agents to reduce traffic to the server. Agents are placed in routers and learn to throttle or restrict traffic to the victim server. This is observed to work well in small-scale network topologies. However, this technique has limited capability in terms of scalability. To eliminate this disadvantage, Malialis and Kudenko [51] proposed the Coordinated Team Learning (CTL) approach that is a new structure of the multi-agent router throttling method based on the divide-and-conquer paradigm. CTL gives a decentralized coordinated response to the DDoS problem. This technique integrates three mechanisms to coordinate or mitigate DDoS attacks that are hierarchical team-based communication, task separation, and team rewards.

Shamshirband et al. [52] studied multi-agent system design to detect intrusion in WSN. They proposed a game method called Game-Fuzzy Q-learning (G-FQL), which combines game theory and fuzzy Q-learning to detect DDoS attacks in WSN. G-FQL is a three-player strategy game for defending against DDoS attacks composed of sink nodes, a base station and an attacker. The game uses the information of past behaviours in the decision-making process of fuzzy Q-learning to detect attacks.

Simpson et al. [53] offer two agent classes created to act on a per-flow basis for any network topology to mitigate DDoS attacks using the RL method. This method is assisted by profound investigation of the availability of the feature and it proves that there are highly predictive flow characteristics for different traffic classes.

Feng et al. [54] used the RL method as a defense method against Application Layer DDoS attacks (L7 DDoS) in their study. Conventional DDoS solutions are difficult to catch and defend against an L7 DDoS attack because the L7 DDoS attack seems legitimate at the transport and network layers. Therefore, a multi-purpose reward function is provided to guide the RL agent. As a result of the study, it was seen that 98.73% of malicious application messages were mitigable. Table 2 summarizes the DoS and DDoS attack.

Table 2 DoS and DDoS attacks.

Reference	**Attack Type**	**Algorithm/Approach**	**Explanation**
Xu et al. [49]	DDoS	HMM	The goal is to separate valid traffic based on source IP addresses from a DDoS attack.
Malialis and Kudenko [50]	DDoS	SARSA	Presented the multi-agent router throttling method by introducing multiple agents to reduce traffic of the server.
Malialis and Kudenko [51]	DDoS	SARSA	Proposed Coordinated Team Learning design on their multiagent router throttling method.
Shamshirband et al. [52]	DDos	Game-Fuzzy Q Learning	Multi-agent system design for detecting intrusions in wireless sensor networks.
Simpson et al. [53]	DDoS	Semi Gradient Sarsa	Train a two agent classes model to drop packets through the analysis on the source destination pair.
Feng et al. [54]	DDoS	MDP	Introduces a multi-objective reward function to guide an RL agent to learn the most suitable action to detect and mitigate AppDDoS attacks

2) Jamming Attacks:

Jamming is a type of attack that works under the principle of broadcasting noise from another station to disrupt a radio broadcast. It can be regarded as a special condition of DoS attacks [4]. Jamming attack has become a severe threat in wireless networks. Different anti-jamming techniques have been improved lately to eliminate this threat [55]. One of the biggest threats to cognitive radio networks (CNR) is jamming attacks [56]. Studies on jamming/anti-jamming in CNR have been done using the RL method [57-62]. In CNR, there are primary users (PU) and secondary users (SU). While PU refers to the users who own the licensed spectrum, SU refers to the unlicensed users who communicate over the licensed spectrum when the PU is not active. Jamming attacks occur in CRNs due to the emergence of smart jammers that can detect jamming frequencies and signal strengths based on the transmission strategies of SUs. Wang et al. [63] developed a game theory

framework to present the interactions of cognitive radio users during a jamming attack. The SU updates their strategy at every stage by observing the status of the channels and the strategy of the attackers from the status of the congested channels. A minimax-Q learning technique is conducted to obtain the optimal anti-jamming channel selection strategy. Using minimax-Q learning, CRN can solve problems regarding to the number of channels and how they switch between various channels to transmit data and control messages. It can also check packets along with channel switching strategy. Lo and Akyildiz [64] proposed jamming-resilient control channel JRCC game to simulate the interaction between cognitive radio users and attackers under the influence of PU. JRCC used user collaboration to facilitate control channel allocations and Win-or-Learn-Fast scheme for jamming resistance in malicious environments. In this scheme, it adapts to PUs activity with learning rates. The optimal control channel allocation strategy for SU is obtained by multi-agent RL.

Xiao et al. [65] studied the interactions between SUs and a smart jammer using game theory. They studied situations where a smart jammer targets to degrade SUs instead of PUs. The Stackelberg equilibrium of the anti-jamming power control game made of a source node, a relay node and a jammer is derived and compared with the Nash equilibrium of the game. Power control strategies with RL techniques like Q-learning and WoLF-PHC are presented to obtain the optimum forces against jamming for SUs without knowing the network parameters.

Han et al. [66] designed a dynamic anti-jamming communication game for CRNs that improves the signal to interference plus noise ratio (SINR) against intelligent jammers. The game represents an environment made up of multiple jammers sending jamming signals to disrupt the SUs' communication. The RL state is the radio medium made of PUs, SUs, jammers and the serving base station. DQN is used as a frequency-hopping policy to see if the SU will exit from a dense jamming area and defeat smart jammers. As a result of the study, it was seen that the anti-jamming system proposed using the DQN algorithm outperforms the Q-learning algorithm with higher SINR, faster convergence rate, lower defense cost and improved use of SU. Liu et al. [67] aimed to enhance Han et al.'s [66] work by proposing an anti-jamming communication system with different and more comprehensive contributions. Instead of using SINR and PU occupancy as in [66], spectrum waterfall using spectrum information with temporal characteristics is used to describe the environmental state. To deal with the infinite state of the spectrum waterfall, a recursive convolutional neural network (RCNN) is modelled and a DRL algorithm for anti-jamming is proposed. The model has been tested with the scenarios of comb jamming, sweeping jamming, dynamic jamming and intelligent comb jamming. The drawback of the studies in [66] and [67] is that they can only acquire the most appropriate policy for one user.

There have been studies on wideband autonomous cognitive radio (WACR) based jamming prevention using RL [68, 69]. Machuzak and Jayaweera [68] studied the design and implementation of WACR for anti-jamming. WACR can acquire spectrum information to locate and identify the sweeping jammer. In this study, the Q-learning method was used to optimally determine a new sub-band that continues to transmit uninterruptedly for a long time when the existing spectrum sub-band is blocked by a jammer. The agent's reward function was determined while it took for the jammer or interferer to interfere with WACR transmission. The results indicate that the

agent can detect jamming patterns and has successfully learned the optimal sub-band selection policy for jammer avoidance. Aref et al. [69] propose the RL method for anti-jamming communication in WACR in a multi-agent environment. WACRs can detect the states of the radio frequency spectrum and the network and autonomously optimize the corresponding operating mode. The aim of every radio is to avoid broadcasts from other WACRs and the sweeping jammer signal affecting the whole spectrum band. The proposed multi-agent RL method is used to avoid crosstalk and interference from other radios by learning the appropriate sub-band selection policy. Their results demonstrate that the proposed multi-agent RL can provide a significant improvement of the anti-jamming protocol against a random policy. Yao and Jia [70] investigated the anti-jamming defense problem in multi-user cases, in which inter-user coordination is considered. The Markov game structure was used to simulate and analyse the anti-jamming defense problem. A collaborative multi-agent anti-jamming (CMAA) algorithm is presented to find out the optimal anti-jamming strategy. CMAA can both solve the external malicious jamming problem and effectively deal with mutual interference between users.

Anti-jamming methods generally depend on frequency jumping to hide or escape jammers. Aforementioned methods are not useful regarding to bandwidth usage and can cause high congestion. Pourranjbar et al. [71] unlike other studies, used a new anti-jamming technique which redirects the jammer to attack a victim channel while legitimate users are communicating on secure channels. Since jammer's channel information is not recognised by users, an optimal channel selection scheme and a suboptimal power allocation algorithm using RL are presented. The efficiency of the proposed method is evaluated by calculating the statistical lower limit of the total received power (TRP). More than 50% of the highest TRP with no jamming for a given access point is achieved when there are a single user and three frequency channels. The presented anti-jamming technique exceeds the compared RL-based anti-jamming methods and the random search method.

Considering the weaknesses of a wireless environment for vehicle communications, both Vehicular Transportation Networks (VANET) and Unmanned Aerial Vehicular (UAV) networks are vulnerable to jamming attacks [72]. Lu et al. [73] and Xiao et al. [74] studied intelligent jamming attacks in a VANET, where a jammer constantly alters its attack strategy taking advantage of UAV devices. To recover vehicle communications, a UAV was used to transmit data to alternate units when roadside units were under jamming attacks. . A game theory technique is used to illustrate the interactions between the jammer and the UAV. Peng et al. [75] studied a communication system in which the communication between the UAV swarm and the base station in compressed by various interventions. Multiple parameters have been considered so that UAV communications can overcome jamming attacks. This study represents a modified Q-Learning algorithm based on multi-parameter programming to provide a balance between the motion and communication performance of UAVs. Li et al. [76] propose an RL method that uses domain information to improve algorithm speed and shrink the state space that the agent has to search. They used signal attenuation in free space and the law of inertia of aircraft to guide the efficient research of UAVs in state space. The subjective value of the task and the performance indicators of the receiver are added to the reward function.

Lu et al. [77] proposed an RL-based robot relay scheme for smart jamming attacks in

UAVs. In this diagram, RL is combined with a functional approach called tile coding. This case is designed to optimize both the robot's travel distance and relay power to improve UAV transmission quality and save robot energy. Robot mobility and relay policyare selected according to signal quality, jamming power, energy consumption and bit error rate of UAV messages. It uses three deep neural networks to select robot mobility and reduce the complexity of transition policy. The structure of the three networks is modelled with fully connected layers rather than convolutional layers. In the proposed scheme, better performance results were obtained than the existing schemes in the probability of interruption, robot energy consumption and bit error rate. Table 3 summarizes the Jamming attacks.

3) Spoofing and Phishing Attack:

A spoofing attack, especially in network security, is a situation in which a person or program is successfully identified as another identity by illegally distorting data. Xiao et al. [78, 79] conducted research on the authentication at the PHY layer in wireless networks. They used RL to detect the spoofing attack and find the optimum test threshold. The interactions between a legitimate receiver and spoofer are modelled as a zero-sum authentication game. Q-learning and Dyna-Q algorithms were used to find the optimum test threshold for spoofing detection.

Benefit states of the receiver or spoofing are calculated grounded on Bayesian risk that is the expected utility in spoofing detection. The receiver targets to choose the most appropriate test threshold in hypothesis testing in PHY layer spoofing detection. Experimental results show that the presented PHY authentication method with RL can improve authentication performance significantly. As a result of the study, it is shown that the proposed PHY authentication technique with RL can enhance authentication performance. Xiao et al. [80] modelled the protection against attacks in smart programmable radio devices of mobile communication throughout the offloading process. The Nash and Stackelberg equilibriums of the offloading game are derived and a Q-Learning based mobile offloading strategy is proposed to enhance the security of mobile devices during offloading. The results show that the presented offloading model can enhance mobile device usage and reduce the attack rate. Radio-based authentication is a procedure to authenticate the device and avoid spoofing attacks. On the other hand, it is not easy to determine the dynamic time variable channel mode in a real environment.

Liu et al. [81] used RL to obtain time-varying channel information. They proposed active authentication of mobile devices with the RL method. PHY layer information is evaluated to detect spoofing attacks. It is accepted that the signal strength received on the receiver side detects the spoofing attack. The receiver calculates the test statistics of the hypothesis test while receiving a packet. If this value is above the threshold value, the receiver detects the packet as a spoofed packet. Q-learning was used to obtain the optimum testing strategy without knowing the incoming packet’s model.

Table 3 Jamming attacks.

Reference	Attack Type	Algorithm/Approach	Explanation
Wang et al. [63]	Jamming	Minimax Q-Learning	They presented a game theory approach to simulate the CRN under jamming attack.
Lo and Akyildiz [64]	Jamming	Win-or-Learn-Fast	The best control channel allocation strategy for SU is derived using multi agent reinforcement learning
Xiao et al. [65]	Jamming	Q-learning and WoLF-PHC	Anti-jamming problem of SU is defined as Stackelberg equilibrium problem.
Liu et al. [67]	Jamming	DQN with CNN	Improved the anti-jamming strategies against dynamic and intelligent jammers
Machuzak and Jayaweera [68]	Jamming	Q-Learning	Q-learning is trained to prevent attacks with a wide range of hundreds of MHz in real time.
Aref et al. [69]	Jamming	Q-Learning	A study of anti-jamming communication in WACR in a multi-agent environment.
Pourranjbar et al. [71]	Jamming	Q-Learning	The goal is to attack the victim channel with a jammer while maintaining users' communications on secure channels.
Lu et al. [73] and Xiao et al., [74]	Jamming	Q-Learning	It is presented to use UAV communications for rerouting traffic from congested areas to alternative RSUs.
Li et al. [76]	Jamming	Q-Learning	Q-learning is modeled to select transport policy to improve SINR with small-scale state space.
Lu et al. [77]	Jamming	Q-learning Safe DDQN	DRL model is proposed to avoid the high interruption hazards of UAV messages using three deep neural networks.

Phishing attacks, on the other hand, is a crime that obtains personal information from users through spoofed websites [82]. In this method, attackers try to obtain people's passwords or credit card information by sending e-mails to individuals as if they are sent from safe sources. Victims who click on the links they send via e-mails are usually directed to spoofed sites and share the information they entered with the attackers. Spoofing is an identity theft in which someone tries to use the identity of a legitimate user. On the other hand, phishing is used to steal a user's sensitive information, such as bank account details. Smadi et al. [83] proposed a phishing detection scheme called a phishing email detection system (PEDS), combining a neural network approach with RL. The proposed model, PEDS, is the first study in this area to use RL to detect zero-day phishing attacks. In

the pre-processing stage, the mail header, email text, URL and HTML content are known as the input to the feature evaluation and reduction algorithm (FERA). The dataset consists of 9118 emails, of which 50% are legitimate and the remainings are phishing emails. It has been observed that the attacks of the proposed system reach high accuracy (98.63%), true positive rate (99.07%) and true negative rate (98.19%) performance levels. In addition, they obtained false positive and false negative rates of 1.81% and 0.93%, respectively. Table 4 summarizes the Spoofing and Phishing attacks.

Table 4 Spoofing and Phishing attacks.

Reference	Attack Type	Algorithm/Approach	Explanation
Xiao et al.[78, 79]	Spoofing	Q-Learning and Dyna-Q	The purpose is selecting the optimum authentication threshold value in wireless networks.
Xiao et al. [80]	Spoofing	Q-Learning	The purpose is to provide security during the offloading process on mobile devices.
Liu et al. [81]	Spoofing	Q-Learning	The purpose is to authenticate mobile devices against attacks.
Smadi et al. [83]	Phishing	Neural network +RL	It is the first study to use RL to determine a zero-day phishing attack.

4) Cross-Site Scripting:

Cross-Site Scripting (XSS) is an attack on a web page through script codes. This attack appears when the developer does not pass the inputs, received from the user, from the necessary HTML and JavaScript filters. When the entries do not pass the necessary filters and at the same time the user is an attacker, it runs malicious codes which are able to harm other users or directly the system. Since HTML, CSS and JavaScript are languages described by XSS, the malicious code can directly harm other users. Fang et al. [84] represent an XSS adversarial attack model based on the RL method (RLXSS). The aim of this study is to optimize the detection of XSS attacks according to adversarial attack models. To achieve this, RL was used to determine the most appropriate escape technique. This RL method demonstrates how detection capabilities against XSS attacks are improved. Tariq et al. [85] used Genetic Algorithms together with RL to deal with XSS attack, which was compared with previous studies. For validation, a real dataset of XSS attacks was used. The proposed approach achieves 99.75% accuracy in the normal state, while it reaches 99.89% accuracy after loading the attacks. The study showed better results as the number of attacks increased. Caturano et al. [86] used the MORL model to generate attack strings that enable the detection of XSS vulnerabilities in web applications. They designed an intelligent agent called Suggester that suggests learned actions to a human upon possible observations. They have trained this agent to generate attack sequences using the MORL environment and action space.

5) SQL Injection:

Recently, many databases are created to conform to commands written in SQL. Many websites acquire information from users and send these data to SQL databases. Attackers take control of victims' databases by exploiting SQL vulnerabilities. Erdodi et al. [87] worked on expressing the SQL injection vulnerability exploit problem using RL. They modelled their work using MDP. It is represented as an attacker or pentester agent,

and the MDP environment as a vulnerable web page with its associated database. Agents assigned with learning policy were deployed to execute SQL injection into the environment. Agents are designed to learn both a certain method to overcome an individual challenge and a more general principle which can be implemented to perform SQL injection attacks on any system. Table 5 summarizes Cross-Site Scripting and SQL Injection attacks.

Table 5 Cross-site scripting and SQL injection attacks.

Reference	Attack Type	Algorithm/Approach	Explanation
Fang et al. [84]	Cross-Site Scripting	Double DQN	Proposes an XSS adversarial attack model formed on the RL method (RLXSS).
Tariq et al. [85]	Cross-Site Scripting	Genetic algorithm +RL	In order to detect XSS attack, GA was used with RL because it gave good results in static analysis.
Caturano et al. [86]	Cross-Site Scripting	Q-Learning	The purpose is to detect XSS vulnerabilities in web applications.
Erdodi et al. [87]	SQL Injection	Q-Learning	The aim is to express the problem of exploiting SQL injection vulnerability using RL.

4. CONCLUSION

The use of RL in the field of cyber security is increasing day by day. In this study, we gathered RL studies in the literature under three headings: penetration testing, IDS and cyberattacks. POMDP, MDP and DQN types of RL algorithms are widely used in Penetration Tests. Studies show that the POMDP model does not scale well for large networks. MDP and DQN algorithms have been presented in the literature both to overcome this limitation of POMDP and to increase the size and performance of the network. In IDS studies, with the use of deep learning and RL together, attacks were detected with higher accuracy. In this study, we discussed the types of DoS, DDoS, Jamming, Spoofing, Phishing, Cross-Site Scripting and SQL injection cyberattacks in which RL is applied. When we consider the studies in general, it is seen that RL algorithms are used more than DRL algorithms. It is clear that DRL will be used more frequently in the future to solve complex and dynamic intrusion detection problems.

Although machine learning for cyber security is a new topic in the literature, it has been observed that the techniques used give promising results for the detection and prevention of attacks. It is expected that RL will develop cyber defense and attack methods and contribute to the reshaping of cyber risks. We expect the RL examined in this study to deal with cyber security problems and offer solutions, lay the foundations for future studies and guide these studies to a large extent.

Funding
The author (s) has no received any financial support for the research, authorship or publication of this study.

The Declaration of Conflict of Interest/ Common Interest
No conflict of interest or common interest has been declared by the author.

The Declaration of Ethics Committee Approval

This study does not require ethics committee permission or any special permission.

The Declaration of Research and Publication Ethics

The author of the paper declares that he complies with the scientific, ethical, and quotation rules of SAUJS in all processes of the paper and that he does not make any falsification on the data collected. In addition, he declares that Sakarya University Journal of Science and its editorial board have no responsibility for any ethical violations that may be encountered and that this study has not been evaluated in any academic publication environment other than Sakarya University Journal of Science.

REFERENCES

[1] B. von Solms, R. von Solms, "Cybersecurity and information security – what goes where?," Information & Computer Security, vol. 26, no. 1, pp. 2–9, 2018.

[2] Z. Guan, J. Li, L. Wu, Y. Zhang, J. Wu, X. Du, "Achieving efficient and secure data acquisition for cloud-supported internet of things in smart grid," IEEE Internet Things Journal, vol. 4, no. 6, pp. 1934–1944, 2017.

[3] J.-H. Li, "Cyber security meets artificial intelligence: a survey," Frontiers of Information Technology & Electronic Engineering, vol. 19, no. 12, pp. 1462–1474, 2018.

[4] T. T. Nguyen, V. J. Reddi, "Deep reinforcement learning for cyber security," arXiv [cs.CR], 2019.

[5] N. D. Nguyen, T. T. Nguyen, H. Nguyen, D. Creighton, S. Nahavandi, "Review, analysis and design of a comprehensive deep reinforcement learning framework," arXiv [cs.LG], 2020.

[6] N. D. Nguyen, T. Nguyen, S. Nahavandi, "System design perspective for human-level agents using deep reinforcement learning: A survey," IEEE Access, vol. 5, pp. 27091–27102, 2017.

[7] M. Riedmiller, T. Gabel, R. Hafner, S. Lange, "Reinforcement learning for robot soccer," Autonomous Robots, vol. 27, no. 1, pp. 55–73, 2009.

[8] K. Mülling, J. Kober, O. Kroemer, J. Peters, "Learning to select and generalize striking movements in robot table tennis," The International Journal of Robotics Research, vol. 32, no. 3, pp. 263–279, 2013.

[9] T. G. Thuruthel, E. Falotico, F. Renda, C. Laschi, "Model-based reinforcement learning for closed-loop dynamic control of soft robotic manipulators," IEEE Transactions on Robotics, vol. 35, no. 1, pp. 124–134, 2019.

[10] I.Arel, C. Liu, T. Urbanik, A. G. Kohls, "Reinforcement learning-based multi-agent system for network traffic signal control," IET Intelligent Transport Systems, vol. 4, no. 2, p. 128, 2010.

[11] J. Jin, C. Song, H. Li, K. Gai, J. Wang, W. Zhang, "Real-time bidding with multi-agent reinforcement learning in display advertising," in Proceedings of the 27th ACM International Conference on Information and Knowledge Management, 2018.

[12] M. E. Taylor, N. Carboni, A. Fachantidis, I. Vlahavas, L. Torrey, "Reinforcement learning agents providing advice in complex video games," Connection Science, vol. 26, no. 1, pp. 45–63, 2014.

[13] C. Amato, G. Shani, "High-level reinforcement learning in strategy games", In AAMAS Vol. 10, pp. 75-82, 2010.

[14] M. Jaderberg, W.M Czarnecki, I. Dunning, L. Marris, G. Lever, A.G. Castaneda, C. Beattıe, N. C. Rabinowıtz, A.S. Morcos, A. Ruderman, N. Sonnerat, T. Green, L. Deason, J. Z. Leibo, D. Silver, D.Hassabis, K. Kavukcuoglu, T. Graepel, "Human-level performance in 3D multiplayer games with population-based reinforcement learning," Science, vol. 364, no. 6443, pp. 859–865, 2019.

[15] T. Liu, B. Huang, Z. Deng, H. Wang, X. Tang, X. Wang, D. Cao, "Heuristics-oriented overtaking decision making for autonomous vehicles using reinforcement learning," IET Electrical Systems in Transportation, vol. 10, no. 4, pp. 417–424, 2020.

[16] W. Gao, A. Odekunle, Y. Chen, Z.-P. Jiang, "Predictive cruise control of connected and autonomous vehicles via reinforcement learning," IET Control Theory Applications, vol. 13, no. 17, pp. 2849–2855, 2019.

[17] F. Richter, R. K. Orosco, M. C. Yip, "Open-sourced reinforcement learning environments for surgical robotics," arXiv [cs.RO], 2019.

[18] C. Shin, P. W. Ferguson, S. A. Pedram, J. Ma, E. P. Dutson, J. Rosen, "Autonomous tissue manipulation via surgical robot using learning based model predictive control," in 2019 International Conference on Robotics and Automation (ICRA), 2019.

[19] H. Snyder, "Literature review as a research methodology: An overview and guidelines". Journal of business research, 104, 333-339, 2019.

[20] P. Davies, "The relevance of systematic reviews to educational policy and practice", Oxford review of education, 26(3-4), 365-378, 2000.

[21] F. M. Zennaro, L. Erdodi, "Modeling penetration testing with reinforcement learning using capture-the-flag challenges and tabular Q-learning", arXiv preprint arXiv:2005.12632, 2020.

[22] R. S. Sutton, A. G. Barto, "Reinforcement Learning: An Introduction", 2nd ed. Cambridge, MA: Bradford Books, 2018.

[23] V. Mnih, V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves,

[24] M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg D. Hassabis., "Human-level control through deep reinforcement learning," Nature, vol. 518, no. 7540, pp. 529–533, 2015.

[25] V. François-Lavet, P. Henderson, R. Islam, M. G. Bellemare, J. Pineau, "An introduction to deep reinforcement learning," Foundations and Trends® in

Machine Learning, vol. 11, no. 3–4, pp. 219–354, 2018.

[26] A.Uprety, D. B. Rawat, "Reinforcement learning for IoT security: A comprehensive survey," IEEE Internet of Things Journal, vol. 8, no. 11, pp. 8693–8706, 2021.

[27] S. P. K. Spielberg, R. B. Gopaluni, P. D. Loewen, "Deep reinforcement learning approaches for process control," in 2017 6th International Symposium on Advanced Control of Industrial Processes (AdCONIP), 2017.

[28] H. Mao, M. Alizadeh, I. Menache, S. Kandula, "Resource management with deep reinforcement learning," in Proceedings of the 15th ACM Workshop on Hot Topics in Networks, 2016.

[29] M. Vecerik T. Hester, J. Scholz, F. Wang, O. Pietquin, B. Piot, N. Heess, R. Thomas,T. Rothörl, T. Lampe, M. Riedmiller, "Leveraging demonstrations for deep Reinforcement Learning on robotics problems with sparse rewards," arXiv [cs.AI], 2017.

[30] S. Gu, E. Holly, T. Lillicrap, S. Levine, "Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates," in 2017 IEEE International Conference on Robotics and Automation (ICRA), 2017.

[31] M. C. Ghanem, T. M. Chen, "Reinforcement learning for intelligent penetration testing," in 2018 Second World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4), 2018.

[32] C. Sarraute, O. Buffet, J. Hoffmann, "POMDPs make better hackers: Accounting for uncertainty in penetration testing," Twenty-Sixth AAAI Conference on Artificial Intelligence, vol. 26, no. 1, pp. 1816–1824, 2021.

[33] C. Sarraute, O. Buffet, J. Hoffmann, "Penetration Testing == POMDP Solving?," arXiv [cs.AI], 2013.

[34] J. Hoffmann, "Simulated penetration testing: From 'Dijkstra' to 'Turing Test++,'" Proceedings of the International Conference on Automated Planning and Scheduling, vol. 25, pp. 364–372, 2015.

[35] J. Schwartz, H. Kurniawati, "Autonomous Penetration Testing using Reinforcement Learning," arXiv [cs.CR], 2019.

[36] M. C. Ghanem, T. M. Chen, "Reinforcement learning for efficient network penetration testing," Information (Basel), vol. 11, no. 1, p. 6, 2019.

[37] A.Chowdhary, D. Huang, J. S. Mahendran, D. Romo, Y. Deng, A. Sabur, "Autonomous security analysis and penetration testing," in 2020 16th International Conference on Mobility, Sensing and Networking (MSN), 2020.

[38] H. Nguyen, S. Teerakanok, A. Inomata, T. Uehara, "The proposal of double agent architecture using actor-critic algorithm for penetration testing," in Proceedings of the 7th International Conference on Information Systems Security and Privacy, 2021.

[39] C. Neal, H. Dagdougui, A. Lodi, J. M. Fernandez, "Reinforcement learning based penetration testing of a microgrid control algorithm," in 2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC), 2021.

[40] Y. Yang, X. Liu, "Behaviour-diverse automatic penetration testing: A curiosity-driven multi-objective deep Reinforcement Learning approach," arXiv [cs.LG], 2022.

[41] X. Xu, T. Xie, "A reinforcement learning approach for host-based intrusion detection using sequences of system calls," in Lecture Notes in Computer Science, Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 995–1003.

[42] S. Aljawarneh, M. Aldwairi, M. B. Yassein, "Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model," Journal of Computational Science, vol. 25, pp. 152–160, 2018.

[43] X. Xu, "Sequential anomaly detection based on temporal-difference learning: Principles, models and case studies," Applied Soft Computing, vol. 10, no. 3, pp. 859–867, 2010.

[44] B. Deokar, A. Hazarnis, "Intrusion detection system using log files and reinforcement learning", International Journal of Computer Applications, vol. 45, no. 19, 28-35, 2012.

[45] S. Otoum, B. Kantarci, H. Mouftah, "Empowering reinforcement learning on big sensed data for intrusion detection," in ICC 2019 - 2019 IEEE International Conference on Communications (ICC), 2019.

[46] G. Caminero, M. Lopez-Martin, B. Carro, "Adversarial environment reinforcement learning algorithm for intrusion detection," Computer Networks, vol. 159, pp. 96–109, 2019.

[47] K. Sethi, E. Sai Rupesh, R. Kumar, P. Bera, Y. Venu Madhav, "A context-aware robust intrusion detection system: a reinforcement learning-based approach," International Journal of Information Security, vol. 19, no. 6, pp. 657–678, 2020.

[48] H. Alavizadeh, H. Alavizadeh, J. Jang-Jaccard, "Deep Q-learning based reinforcement learning approach for network intrusion detection," Computers, vol. 11, no. 3, p. 41, 2022.

A. S. S. Alawsi, S. Kurnaz, "Quality of service system that is self-updating by intrusion detection systems using reinforcement learning," Applied Nanoscience, 2022.

[49] X. Xu, Y. Sun, Huang, Z, "Defending DDoS attacks using hidden Markov models and cooperative reinforcement learning", In Pacific-Asia Workshop on Intelligence and Security Informatics (pp. 196-207). Springer, Berlin, Heidelberg, 2007.

[50] K. Malialis, D. Kudenko, "Multiagent Router Throttling: Decentralized coordinated response against DDoS attacks," In Twenty-Fifth IAAI Conference, vol. 27, no. 2, pp. 1551–1556, 2013.

[51] K. Malialis, D. Kudenko, "Distributed response to network intrusions using

multiagent reinforcement learning," Engineering Applications of Artificial Intelligence, vol. 41, pp. 270–284, 2015.

[52] S. Shamshirband, A. Patel, N. B. Anuar, M. L. M. Kiah, A. Abraham, "Cooperative game theoretic approach using fuzzy Q-learning for detecting and preventing intrusions in wireless sensor networks," Engineering Applications of Artificial Intelligence, vol. 32, pp. 228–241, 2014.

[53] K. A. Simpson, S. Rogers, D. P. Pezaros, "Per-host DDoS mitigation by direct-control reinforcement learning," IEEE Transactions on Network and Service Management, vol. 17, no. 1, pp. 103–117, 2020.

[54] Y. Feng, J. Li, T. Nguyen, "Application-layer DDoS defense with reinforcement learning," in 2020 IEEE/ACM 28th International Symposium on Quality of Service (IWQoS), 2020.

[55] K. Grover, A. Lim, Q. Yang, "Jamming and anti-jamming techniques in wireless networks: a survey," International Journal of Ad Hoc and Ubiquitous Computing, vol. 17, no. 4, p. 197, 2014.

[56] Y. Wu, B. Wang, K. J. R. Liu, T. C. Clancy, "Anti-jamming games in multi-channel cognitive radio networks," IEEE journal on selected areas in communications, vol. 30, no. 1, pp. 4–15, 2012.

[57] S. Singh, A. Trivedi, "Anti-jamming in cognitive radio networks using reinforcement learning algorithms," in 2012 Ninth International Conference on Wireless and Optical Communications Networks (WOCN), 2012.

[58] Y. Gwon, S. Dastangoo, C. Fossa, H. T. Kung, "Competing Mobile Network Game: Embracing antijamming and jamming strategies with reinforcement learning," in 2013 IEEE Conference on Communications and Network Security (CNS), 2013.

[59] K. Dabcevic, A. Betancourt, L. Marcenaro, C. S. Regazzoni, "A fictitious play-based game-theoretical approach to alleviating jamming attacks for cognitive radios," in 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2014.

[60] F. Slimeni, B. Scheers, Z. Chtourou, V. Le Nir, "Jamming mitigation in cognitive radio networks using a modified Q-learning algorithm," in 2015 International Conference on Military Communications and Information Systems (ICMCIS), 2015.

[61] F. Slimeni, B. Scheers, Z. Chtourou, V. Le Nir, R. Attia, "Cognitive radio jamming mitigation using Markov decision process and reinforcement learning," Procedia Computer Science, vol. 73, pp. 199–208, 2015.

[62] F. Slimeni, B. Scheers, Z. Chtourou, V. L. Nir, R. Attia, "A modified Q-learning algorithm to solve cognitive radio jamming attack," International Journal of Embedded Systems, vol. 10, no. 1, p. 41, 2018.

[63] B. Wang, Y. Wu, K. J. R. Liu, T. C. Clancy, "An anti-jamming stochastic game for cognitive radio networks," IEEE journal on selected areas in communications, vol. 29, no. 4, pp. 877–889, 2011.

[64] B. F. Lo, I. F. Akyildiz, "Multiagent jamming-resilient control channel game for cognitive radio ad hoc networks," in 2012 IEEE International Conference on Communications (ICC), 2012.

[65] L. Xiao, Y. Li, J. Liu, Y. Zhao, "Power control with reinforcement learning in cooperative cognitive radio networks against jamming," The Journal of Supercomputing, vol. 71, no. 9, pp. 3237–3257, 2015.

[66] G. Han, L. Xiao, H. V. Poor, "Two-dimensional anti-jamming communication based on deep reinforcement learning," in 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2017.

[67] X. Liu, Y. Xu, L. Jia, Q. Wu, A. Anpalagan, "Anti-jamming communications using spectrum waterfall: A deep reinforcement learning approach," IEEE Communications Letters, vol. 22, no. 5, pp. 998–1001, 2018.

[68] S. Machuzak, S. K. Jayaweera, "Reinforcement learning based anti-jamming with wideband autonomous cognitive radios," in 2016 IEEE/CIC International Conference on Communications in China (ICCC), 2016.

[69] M. A. Aref, S. K. Jayaweera, S. Machuzak, "Multi-agent reinforcement learning based cognitive anti-jamming," in 2017 IEEE Wireless Communications and Networking Conference (WCNC), 2017.

[70] F. Yao, L. Jia, "A collaborative multi-agent reinforcement learning anti-jamming algorithm in wireless networks," IEEE wireless communications letters, vol. 8, no. 4, pp. 1024–1027, 2019.

[71] Pourranjbar, G. Kaddoum, A. Ferdowsi, W. Saad, "Reinforcement learning for deceiving reactive jammers in wireless networks," IEEE Transactions on Communications, vol. 69, no. 6, pp. 3682–3697, 2021.

[72] H. Pirayesh, H. Zeng, "Jamming attacks and anti-jamming strategies in wireless networks: A comprehensive survey," arXiv [cs.CR], 2021.

[73] X. Lu, D. Xu, L. Xiao, L. Wang, W. Zhuang, "Anti-jamming communication game for UAV-aided VANETs," in GLOBECOM 2017 - 2017 IEEE Global Communications Conference, 2017.

[74] L. Xiao, X. Lu, D. Xu, Y. Tang, L. Wang, W. Zhuang, "UAV relay in VANETs against smart jamming with reinforcement learning," IEEE Transactions on Vehicular Technology, vol. 67, no. 5, pp. 4087–4097, 2018.

[75] J. Peng, Z. Zhang, Q. Wu, B. Zhang, "Anti-jamming communications in UAV swarms: A reinforcement learning approach," IEEE Access, vol. 7, pp. 180532–180543, 2019.

[76] Z. Li, Y. Lu, X. Li, Z. Wang, W. Qiao, Y. Liu, "UAV networks against multiple maneuvering smart jamming with knowledge-based reinforcement learning," IEEE Internet of Things Journal, vol. 8, no. 15, pp. 12289–12310, 2021.

[77] X. Lu, J. Jie, Z. Lin, L. Xiao, J. Li, Y. Zhang, "Reinforcement learning based energy efficient robot relay for unmanned aerial vehicles against smart jamming," Science China Information Sciences, vol. 65, no. 1, 2022.

[78] L. Xiao, Y. Li, G. Liu, Q. Li, W. Zhuang, "Spoofing detection with reinforcement learning in wireless networks," in 2015 IEEE Global Communications Conference (GLOBECOM), 2015.

[79] L. Xiao, Y. Li, G. Han, G. Liu, W. Zhuang, "PHY-layer spoofing detection with reinforcement learning in wireless networks," IEEE Transactions on Vehicular Technology, vol. 65, no. 12, pp. 10037–10047, 2016.

[80] L. Xiao, C. Xie, T. Chen, H. Dai, H. V. Poor, "A mobile offloading game against smart attacks," IEEE Access, vol. 4, pp. 2281–2291, 2016.

[81] J. Liu, L. Xiao, G. Liu, Y. Zhao, "Active authentication with reinforcement learning based on ambient radio signals," Multimedia Tools and Applications, vol. 76, no. 3, pp. 3979–3998, 2017.

[82] S. Purkait, "Phishing counter measures and their effectiveness – literature review," Information Management & Computer Security., vol. 20, no. 5, pp. 382–420, 2012.

[83] S. Smadi, N. Aslam, L. Zhang, "Detection of online phishing email using dynamic evolving neural network based on reinforcement learning," Decision Support Systems, vol. 107, pp. 88–102, 2018.

[84] Y. Fang, C. Huang, Y. Xu, Y. Li, "RLXSS: Optimizing XSS detection model to defend against adversarial attacks based on reinforcement learning," Future internet, vol. 11, no. 8, p. 177, 2019.

[85] Tariq, M. A. Sindhu, R. A. Abbasi, A. S. Khattak, O. Maqbool, G. F. Siddiqui, "Resolving cross-site scripting attacks through genetic algorithm and reinforcement learning," Expert Systems with Applications., vol. 168, no. 114386, p. 114386, 2021.

[86] F. Caturano, G. Perrone, S. P. Romano, "Discovering reflected cross-site scripting vulnerabilities using a multiobjective reinforcement learning environment," Computers & Security, vol. 103, no. 102204, p. 102204, 2021.

[87] L. Erdodi, Å. Å. Sommervoll, F. M. Zennaro, "Simulating SQL injection vulnerability exploitation using Q-learning reinforcement learning agents," arXiv [cs.CR], 2021.