Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.34, NO.1, Feb. 2024 ISSN 2288-2715(Online)
https://doi.org/10.13089/JKIISC.2024.34.1.21

강화학습 기반 네트워크 취약점 분석을 위한 적대적 시뮬레이터 개발 연구*

김 정 윤,[1†] 박 종 열,[2] 오 상 호[3‡]
[1,2]서울과학기술대학교 (대학원생, 교수), [3]국립부경대학교 (교수)

A Study on the Development of Adversarial Simulator for Network Vulnerability Analysis Based on Reinforcement Learning*

Jeongyoon Kim,[1†] Jongyoul Park,[2] Sang Ho Oh[3‡]
[1,2]Seoul National University of Science & Technology (Graduate student, Professor),
[3]Pukyong National University (Professor)

요 약

ICT와 network의 발달로 규모가 커진 IT 인프라의 보안 관리가 매우 어려워지고 있다. 많은 회사나 공공기관에서 시스템과 네트워크 보안 관리에 어려움을 겪고 있다. 또한 하드웨어와 소프트웨어의 복잡함이 커짐에 따라 사람이 모든 보안을 관리한다는 것은 불가능에 가까워지고 있다. 따라서 네트워크 보안 관리에 AI가 필수적이다. 하지만 실제 네트워크 환경에 공격 모델을 구동하는 것은 매우 위험하기에 실제와 유사한 네트워크 환경을 구현하여 강화학습을 통해 사이버 보안 시뮬레이션 연구를 진행하였다. 이를 위해 본 연구는 강화학습을 네트워크 환경에 적용하였고, 에이전트는 학습이 진행될수록 해당 네트워크의 취약점을 정확하게 찾아냈다. AI를 통해 네트워크의 취약점을 발견하면, 자동화된 맞춤 대응이 가능해진다.

ABSTRACT

With the development of ICT and network, security management of IT infrastructure that has grown in size is becoming very difficult. Many companies and public institutions are having difficulty managing system and network security. In addition, as the complexity of hardware and software grows, it is becoming almost impossible for a person to manage all security. Therefore, AI is essential for network security management. However, since it is very dangerous to operate an attack model in a real network environment, cybersecurity emulation research was conducted through reinforcement learning by implementing a real-life network environment. To this end, this study applied reinforcement learning to the network environment, and as the learning progressed, the agent accurately identified the vulnerability of the network. When a network vulnerability is detected through AI, automated customized response becomes possible.

Keywords: Reinforcement Learning, Information Security, Network, DQN

I. 서 론

정보통신기술과 네트워크의 규모가 커짐에 따라 IT인프라는 더욱 복잡해지고 관리하기 어려워지고 있다. 복잡한 IT인프라의 보안 관리는 수동으로 모든 작업을 처리하는 것이 불가능하기에 AI를 도입하

Received(09. 05. 2023), Modified(11. 02. 2023), Accepted(12. 06. 2023)
* 이 성과는 정부(과학기술정통신부)의 재원으로 일부 한국연구재단의 지원을 받아 수행(No. RS-2023-00221365)하고 일부 정보통신기획평가원의 지원을 받아 수행(No.2021-0-00796) 연구입니다.
† 주저자, kjy97426@seoultech.ac.kr
‡ 교신저자, shoh0320@pknu.ac.kr(Corresponding author)

여 자동으로 처리해야 한다[1]. 더욱이 자율적으로 동작하는 멀웨어, 취약성을 악용하여 감염을 확산시키는 워너크라이와 같은 새로운 형태의 사이버 공격에 의한 위협도 같이 증가하고 있어, 대규모 피해가 계속 증가하고 있다[2].

실제 네트워크 환경에서 공격모델을 작동시켜 취약점을 발견하기엔 위험하고, 불가능하기에 실제와 유사한 네트워크 환경을 구성하여 실험을 진행하는 연구가 활발히 이루어지고 있다. 기존의 정보보안시스템의 한계로 인공지능 기술을 활용하여 사업 공격에 대응하는 연구[3], 과거의 이상 상태 데이터를 학습하여 보안 위협 감지 및 예측을 하는 인공지능 기반 기술[4], 머신러닝과 딥러닝 모델을 이용해 사이버 보안 공격자를 예측하는 기술[5] 등 인공지능 기술 기반 보안 연구가 많이 진행되고 있다.

하지만 기존 AI기반의 정보보호기술은 많은 한계점을 지니고 있다. 네트워크의 이상탐지를 통해 보안 시스템을 구축하는 경우, SVM, Decision Tree, random forest 등의 알고리즘을 사용하는데, 복잡하고, 만족스럽지 못한 가정의 데이터가 있을 경우 이러한 분류 알고리즘들은 좋은 성능을 발휘하지 못했다[6]. 신경망을 사용하는 딥러닝 모델의 경우 변이된 데이터가 입력으로 들어올 때, 평균보다 못한 성능을 보이는 경우가 많았다[7]. 즉, 기존의 머신러닝 알고리즘과 딥러닝 모델은 변이에 대한 대응력이 부족하다는 단점이 있음을 알 수 있다. 이에 본 연구는 강화학습 기반 사이버 보안 연구를 진행하였다.

대부분의 기존 강화학습 기반 사이버 보안 연구는 마이크로소프트에서 제공하는 CyberBattleSim과 같은 가상의 시뮬레이터를 기반으로 진행되었다[8]. 이러한 가상 시뮬레이터의 경우 현실과 맞지 않은 부분이 많아 재현성이 떨어진다고 볼 수 있다. 따라서 본 연구는 실제로 일어났던 해킹 공격 시나리오를 가져와 실험을 진행하였다.

실제 네트워크로 연결된 분산 환경의 공격은 state space가 매우 크기 때문에 제약을 둘 수 밖에 없다[9]. 그래서 매우 제한적인 연구만 가능했다. 또한 자동화되고, AI로 대체된 해커들이 많아져 한정된 인적 자원으로 완벽한 보안을 확신할 수 없어졌다[10][11]. 따라서 실제 환경에서 공격 모델을 실행했을 때의 위험성과 제한적인 가상 공간에서 진행되는 실험의 제약, 자동화된 공격 시나리오를 다른 도메인으로 전이하는 연구의 필요성이 높아졌다. 본 연구는 MITRE ATT&CK 데이터베이스에서 가져온, 실제 있었던 사이버 공격 시나리오를 가상 환경에서 구현하여 강화학습을 이용해서 해당 네트워크의 취약점이나 효과적인 사이버 공격을 찾고, 자동화된 네트워크 보안을 구성하는 것에 목적이 있다.

본 연구에서는 강화학습 기반의 소규모 데이터에서 학습 및 공격 시나리오 탐지 기술을 개발한다. 또한 마이크로소프트에서 개발한 CyberBattleSim을 바탕으로 베이스라인 환경을 구축하고, 강화학습 기법을 적용하여 적대적 시뮬레이션을 가능하게 하는 프로토타입 기술을 개발한다.

이번 시뮬레이터 개발을 통해 대규모 네트워크 상황에서 운영이 어려운 다양한 공격 시뮬레이션의 시험 운영이 가능해질 것이다. 또한 다양한 변이 공격에 대한 모의시험으로 실제 네트워크에는 영향을 주지 않지만, 매우 정확하고 현실과 유사한 연구를 진행할 수 있을 것이다. 또한 적은 학습 데이터와 많은 변이가 생성되는 상황에서도 동작하는 강화학습 기반 탐지 기술을 확보할 수 있을 것이다.

본 논문은 배경 이론과 사용한 시나리오, 시뮬레이션 환경 구현 및 실험 결과, 결론으로 구성되어 있다.

II. 배경 이론

2.1 강화학습

머신러닝은 학습방법에 따라 크게 3가지로 나뉜다. 학습 데이터와 label이 주어지는 지도학습(supervised learning), label없이 학습 데이터만 주어지는 비지도학습(unsupervised learning), 순차적 의사 결정문제에서 누적 보상을 최대화하기 위해 시행착오를 통해 행동을 교정하는 학습 과정인 강화학습(reinforcement learning)이 있다[12].

Fig. 1.에서 알 수 있듯이, 강화학습은 agent가 environment와 상호작용하여 얻은 누적 보상의 최대화를 목표로 한다. 이 때, agent가 보고 판단할

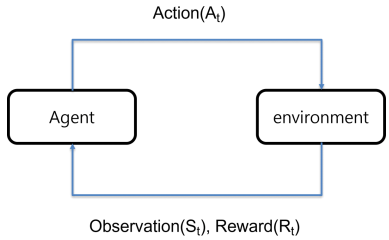

Fig. 1. Reinforcement Learning Progress

수 있는 현재 상태를 state, agent가 관측한 상태를 observation, agent가 선택하여 실행할 수 있는 행동을 action, environment와 상호작용하여 얻은 보상을 reward라고 한다.

강화학습을 진행한다는 것은 위의 순차적 의사 결정문제를 해결한다는 것이다. 순차적 의사 결정문제는 Markov Decision Process(MDP)로 모델링 될 수 있다. MDP란 로 바로 이전의 state만이 다음 state에 영향을 미칠 수 있는 결정 과정이다. MDP는 총 5가지의 파라미터로 정의될 수 있다. 상태 집합 S(state), 행동 집합 A(action), 전이확률 행렬(probability), 보상함수(reward), 감쇠인자(gamma)로 정의된다. 상태 집합는 가능한 상태를 모두 모아놓은 집합이다. 행동 집합은 agent가 할 수 있는 action들을 모두 모아놓은 집합이고, 전이확률 행렬은 어떤 state S에서 action을 하나 선택 및 실행했을 때, state S'로 갈 수 있는 확률을 행렬로 나타낸 것이다. 보상함수는 state S에서 action을 통해 state S'에 도착했을 때, 받는 보상을 계산해 리턴해주는 함수이다. 강화학습에서는 이 보상함수를 통해 나온 reward들의 누적된 총합이 최대가 되게 하는 것이 목표이다. 감쇠인자는 미래에서 얻을 보상이 현재 얻을 보상과 비교했을 시 상대적으로 얼마나 중요한가를 계산할 수 있게 해주는 파라미터이다.

강화학습은 agent와 environment가 서로 상호작용하여 cummulative reward가 최대가 되는 policy를 구하는 것이 주된 목표이다. 강화학습이 가장 잘 적용된 분야인 게임을 예시로 들어보면, 게임 플레이어는 agent, 명령은 action, 게임 내부는 environment, 이기거나 임무 완수 시 받는 양의 reward 등의 게임 특징이 강화학습이 적용되기 쉽게 만들어준다. 이와 비슷한 맥락으로 마이크로소프트는 강화학습을 소프트웨어 보안에 적용시켜 보았다. 해킹의 공격자는 agent에, 해당 컴퓨터의 특성을 state, 해킹의 기술을 action, 해킹의 대상 네트워크를 environment, 해킹 성공 여부를 reward에 각각 담을 수 있기 때문이다.

agent와 environment가 상호작용할 때, 시작 상태 S_0에서 종료 상태 S_T까지의 시퀀스를 에피소드라고 한다. 에피소드 안에는 여러 스텝이 있고, 한 스텝은 state, action, reward, next state로 이루어져 있다. 여러 번의 agent와 environment의 상호작용으로 나온 에피소드 샘플들을 모아 agent를 학습시키는 데이터로 이용한다.

2.2 CyberBattleSim

CyberBattleSim은 마이크로소프트에서 개발한 모의 네트워크 보안 시뮬레이션 연구 toolkit이다. 실제 컴퓨터와 시스템, 애플리케이션을 시뮬레이션으로 구현하기 매우 어렵고, 불가능에 가깝기 때문에 CyberBattleSim은 컴퓨터 네트워크와 보안 시스템을 높은 수준으로 추상화시켰다. 그렇기 때문에 stack buffer overflow와 같은 low level 해킹의 관점이 아닌 어떤 네트워크에 대한 전반적인 공격의 흐름을 이해하기위해 개발되었다[13].

Table 1., Table 2.에서 알 수 있듯이 CyberBattleSim은 컴퓨터 노드로 구성된 네트워크를 environment로 설정한다. 공격자이자 agent인 attacker는 컴퓨터 노드를 공격하기 위해 local attack, remote attack, authenticated connection 총 3가지 action을 선택할 수 있다. reward는 점령한 노드의 가치로 매겨진다. state는 컴퓨터 노드로 설정된다.

Table 1. Mapping Reinforcement Learning to Cybersecurity1

Observation space	Reward
Discovered nodes Owned nodes Discovered credentials Escalation levels Available attacks	Intrinsic node value

Table 2. Mapping Reinforcement Learning to Cybersecurity2

Environment	Action space
State = network Single-agent Partially observable Deterministic Static Discrete Post-breach	Local attack Remote attack Authenticated connection

2.3 MITRE ATT&CK

MITRE ATT&CK은 실재했던 해킹이나 공격들

의 전술 및 기술을 모아놓은 데이터베이스를 제공하는 비영리 단체이다. 지금까지 쌓아놓은 데이터를 기반으로 보안 프레임워크를 제공한다. MITRE ATT&CK은 공격 전략인 tactics와 공격 기술인 techniques 두 종류로 분류해놓았고, 그 안에서 다시 대상을 기준으로 enterprise와 mobile, Industrial Control System로 분류해놓았다. 전체 tactics와 techniques을 표로 볼 수 있고, 실제 있었던 공격 시나리오에서 썼던 공격 전략 및 기술들을 한 번에 볼 수 있다[14].

MITRE ATT&CK의 공격 전략인 tactics는 공격 목적들을 기준으로 공격 기술들을 모아둔 카테고리이다. 사이버 공격은 기본적으로 록히드 마틴에서 발표한 정찰, 무기준비, 전달, 공격, 설치, 명령 제어, 목표달성으로 이루어진 사이버 킬체인 7 단계로 이루어져 있다[15]. MITRE ATT&CK의 tactics는 이 7단계를 기반으로 공격자의 techniques이 분류되어있다. 총 40개의 tactics 안에 392개의 techniques이 있고, 실제 있었던 시나리오를 선택하면 tactics별로 어떤 techniques이 쓰였는지 자세히 나온다.

III. 시뮬레이션 실험 및 제안

3.1 MITRE ATT&CK 시나리오 구성

본 연구는 실재했던 사이버 공격의 시나리오를 그 공격에 쓰였던 tactics와 techniques을 구현해 시뮬레이션을 진행하였다. MITRE ATT&CK의 Ajax Security Team의 techniques을 action으로 구현하고, 프로세스를 가져와 시뮬레이션 시나리오를 구성하였다.

Table 3. Lockheed Martin's Cyber Kill Chain Step7

1	Reconnaissance	Invetigate attack targets
2	Weaponization	Cyber Weapon Preparation
3	Delivery	Cyber Weapon Delivery
4	Exploitation	Cyber Weapon Operation
5	Installation	Install on target system
6	Command & Control	Obtaining permissions on the target system
7	Action on Objectives	Goal

Ajax Security team의 공격은 한 때, 미국에 매우 위협적인 영향을 미쳤다[16]. 따라서 테스트 시나리오 적합하여 본 시나리오를 선택했다.

Ajax Security team이 접근에 사용한 techniques에는 spearphishing, credential from web browsers, key logging 총 3가지가 있다. 본 연구에서는 앞의 공격 3가지에 흔히 사이버 공격에 많이 쓰이는 port scan까지 총 4가지를 attacker가 할 수 있는 action으로 설정하였다.

3.2 시뮬레이션 환경 구현

본 연구는 딥러닝 프레임워크인 PyTorch를 사용하고, 시뮬레이션 환경 구현을 위해 사용할 PC와 attacker, environment class를 정의하였다. 본 연구의 실험코드는 CyberBattleSim의 코드의 내부 구성을 참고하여 모두 새롭게 작성되었다. CyberBattleSim은 client와 server를 나타내는 노드들이 존재하고, attacker가 action을 통해 하나씩 점령한다. 따라서 본 연구도 이와 같이 노드들을 정의하고, attacker가 action을 통해 목적을 달성하는 방식으로 진행되었다. CyberBattleSim의 큰 그림만 가져오고, 노드들의 state, attacker의 action등은 모두 본 연구를 위해 새로 구현되었다.

시뮬레이션 네트워크 환경을 구성하는 주요 요소 중 첫 번째인 PC 객체는 본인의 관리자 권한을 얻을 수 있는 credential과 방화벽으로 막혀있는 port들, 서로의 관리자 권한을 credential 없이 얻을 수 있는 허가된 PC목록, 본인과 연결되어 있는 PC들의 credential을 유출할 수 있는 취약점을 가지고 있다. 키보드 보안이 없을 때 key logging을 당할 수 있고, web에 credential이 남아 있을 수도 있다. 악성 mail을 열었을 때, 특정 확률로 바이러스에 감염될 수 있다.

두 번째 요소인 attacker는 제일 먼저 해당 PC 객체의 취약점을 선택해 공격하는 exploit attack과 열려있는 포트를 찾는 port scan을 사용해 목표 PC 객체를 공격한다. attacker는 앞의 공격으로 찾은 연결된 노드 개수와 opened port의 개수, 키보드 보안의 유무, web credential의 유무 총 4가지의 속성을 state로 두고, 현 state에서 가장 합리적인 action을 선택해 목표 PC를 공격한다. attacker가 할 수 있는 action에는 4가지가 있다. 열려있는 port를 통해 접근할 수 있는 try port

access, 허가된 PC의 ip로 둔갑하여 접속하는 spoofing login, 키보드 보안이 없을시 키보드 입력을 빼돌려 credential을 얻는 key logging, web에 저장되어있는 credential을 가져오는 access_web이 있다. 먼저 try port access는 노드들의 port개수를 65534개로 설정하고, 접근 가능한 port들과 방화벽에 의해 접근 불가능한 port들을 분리한 뒤, 접근 가능한 port들에 접근하는 방식으로 구현되었다. 이 또한 잠겨있는 port와 열려있는 port로 구성하여 성공과 실패를 나눌 수 있게 구현하였다. spoofing login은 앞선 공격에 점령당한 노드에 다른 노드에 접근할 수 있는 권한이 있는 경우 해당 노드의 ip로 둔갑하여 접근 가능하도록 구현하였다. key logging은 키보드 보안 프로그램이 설치되지 않은 노드들에 한 해 attacker가 credential을 빼돌릴 수 있게 구현했고, web 응용프로그램에 credential이 남아있는 경우 attacker가 가져올 수 있도록 access_web이 구현되었다. 각 PC 노드들이 키보드 보안 프로그램을 가질 경우나 web 응용프로그램에 credential이 남아있는 경우 모두 환경 초기화 과정에서 확률적으로 발생하게 구현하였다. key logging〔표 4〕에 구현한 MDP가 정리되어있다.

세 번째 요소인 environment 객체는 random seed에 따라 PC객체의 특성과 네트워크 구성을 초기화해준다. attacker가 policy에 따라 공격하고, PC의 credential이 탈취당하는 모든 상황 환경을 제공한다.

강화학습 알고리즘은 크게 value-based 에이전트와 policy-based 에이전트로 분류할 수 있다.

본 연구에서는 대표적인 value-based 강화학습 알고리즘인 Q-learning과 성능이 더 향상된 Deep Q-Network(DQN)을 사용했다. 정책기반 강화학습 알고리즘으로는 Actor-Critic과 Proximal Policy Optimization(PPO)를 사용하였다.

Table 4. MDPs in Emulation Environments Environment

MDP	Contents
State	4 Dimensions tuple ① No. of opened ports ② No. of allowed and connected pc ③ key_sercurity ④ Web_credential
Action	① Opened port attack ② Spoogfing ③ key_logging ④ Access_web
Reward	① attack success: +1 ② attack failed: -1

3.3 시뮬레이션 프레임워크

시뮬레이션 네트워크 환경 프로세스는 다음과 같이 진행된다. Fig. 2.에 나오는 것처럼, 먼저 attacker는 공격할 대상을 탐색한다. 해당 PC에 해당하는 올바른 취약점 아이디를 가지고 공격에 성공한다면 attacker는 발견된 노드에 접속할 수 있는 credential을 얻게 되고, 실패하면 연결된 노드만 발견하고 credential은 얻지 못한다. attacker는 발견된 노드를 공격하기 전 이 프로세스들을 먼저 적용하여, 해당 네트워크에 존재하는 다른 노드를 발견하거나 credential을 획득한다.

Fig. 3.에 나오는 것처럼, 노드 발견 이후 해당 credential을 가지고 있는 때는 추가 공격없이 바로 점령한다. 하지만 해당 노드의 credential이 없을 경우 attacker는 포트스캔 공격을 통해 opened port들을 찾는다. open port들을 찾은 후,

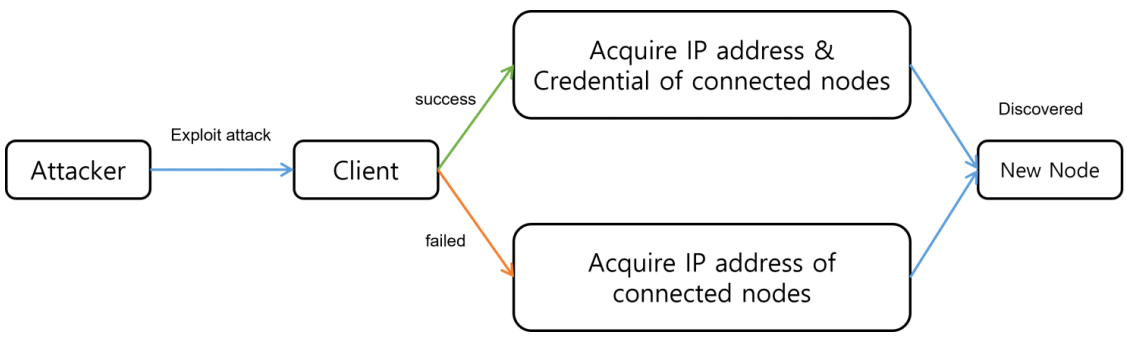

Fig. 2. Exploited node discovery process

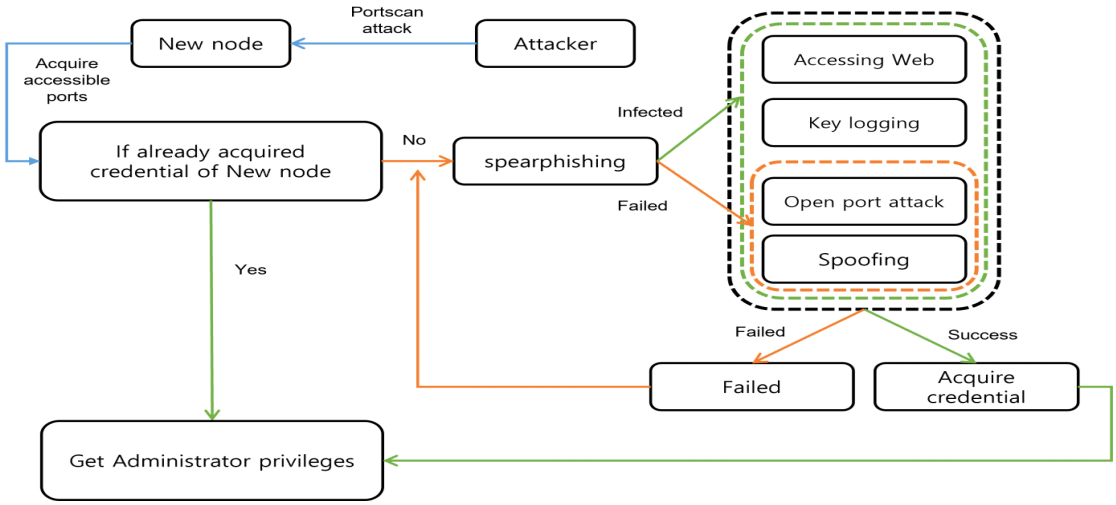

Fig. 3. Node attack process

spearphising, open port attack, key logging, credential from web browser 총 4개의 action들을 credential을 얻을 때까지 번갈아가며 반복적으로 시도한다. 공격이 성공하여 credential을 얻게 되면 attacker는 해당 노드를 점령하게 된다.

시뮬레이션 네트워크 환경에서 attacker가 한 PC를 공격해 Administrator privileges를 성공적으로 확보하게 되면 점령한 것으로 정의했고, 성공적으로 점령하면 +1의 reward를, 실패했을 경우는 -1의 reward를 받게 된다.

IV. 시뮬레이션 결과

실험 결과는 학습 방법에 따라 분류하여 그래프로 나타내었다.

실험에는 신경망을 사용하는 DQN, Actor Critic, PPO 강화학습 알고리즘을 사용하였다. 베이스라인 강화학습 알고리즘으로는 신경망을 쓰지 않는 tabular Q-learning을 사용하였다.

Fig. 4.은 구성된 네트워크 환경에서의 강화학습 알고리즘에 따른 cumulative reward를 나타낸 것이다. 학습이 진행될수록 attacker가 최종적으로 얻는 cumulative reward가 서서히 증가함을 알 수 있다. attacker는 4가지 action 모두 target node를 점령하기 위해 시도해볼 것이며, 학습이 진행됨에 따라 target node 점령에 가장 확률이 높은 action을 선택하여 공격할 것이다.

베이스라인 알고리즘인 tabular Q-learning은 신경망을 쓰지 않고 table에 기록하는 방식인 Dynamic Programming 덕분에 신경망 기반의 다른 강화학습 알고리즘들보다 초기 성능이 좋다. 하지만 데이터가 쌓이고, 학습이 진행되자 곧바로 다른 알고리즘들에 비해 성능이 따라 잡히는 것을 볼 수 있다. 또한 state와 action space가 커지면 긴 학습 시간과 큰 메모리가 필요하기 때문에[17] 다른 알고리즘에 비해 학습이 매우 불안정함을 볼 수 있다. DQN은 soft update를 썼음에도 불구하고 160episode를 넘어가자 local minima에 빠진 것을 볼 수 있다. local minima에 빠지게 되면, 그곳을 탈출하지 않는 한 학습이 계속되어도 그 정도 성능이 계속 유지된다. Actor-Critic의 누적 보상이 가장 안정적이고, 높게 올라간 것을 볼 수 있다. Policy-based 알고리즘인 Actor-Critic이 value-based 알고리즘인 Q-learning이나 DQN

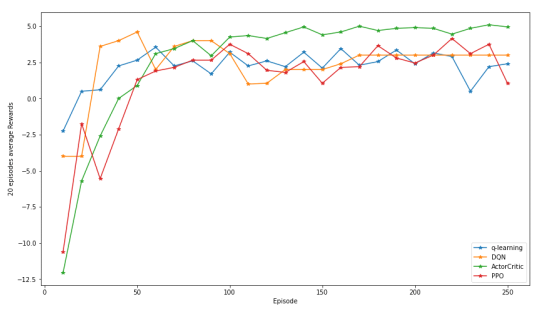

Fig. 4. 20 episodes average rewards

보다 좀 더 유연한 선택을 할 수 있기 때문에 시뮬레이션 실험에서는 강력한 모습을 보인다.

Fig. 5.은 episode에 따른 성공확률을 나타낸 그래프이다. 본 연구에서는 학습할 네트워크 환경의 모든 PC를 점령하면 terminal state로 이동해 해당 에피소드가 종료된 것으로 보고 시뮬레이션을 진행했다.

성공확률 그래프 또한 누적 보상 그래프와 비슷한 경향을 보였다. 성공확률이 높을수록 누적 보상도 높기 때문에 알고리즘별 그래프가 비슷하게 나왔다. DQN의 초기 학습속도가 다른 알고리즘에 비해 매우 빠른 이유는 액션의 선택에 있어서 결정론적이기 때문이다. DQN은 학습 중 일 때, 1-ε의 확률로 가장 높은 Q값을 가지고 있는 action을 선택하기 때문에 조금만 학습이 진행되어도 높은 성능을 보였다. 하지만 local minima에 빠지면서 성능의 한계를 보였다.

Fig. 6.은 학습이 진행됨에 따른 iteration의 개수변화를 보여주고 있다. 해당 노드를 점령하기 위한 시도가 점차 줄어들고 빠르게 점령을 성공하는 것을 볼 수 있다. 성능이 가장 좋은 Actor-Critic의 시도 횟수가 가장 적은 것을 볼 수 있다.

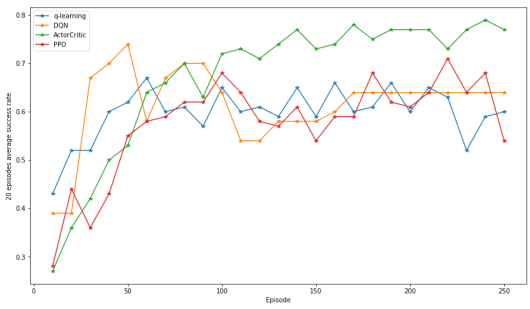

Fig. 5. 20 episodes average success rate

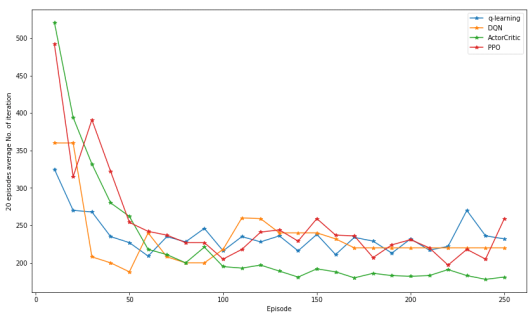

Fig. 6. 20 episodes average No. of iterations

local minima에 빠진 DQN은 iteration의 개수도 변함이 없음을 알 수 있다. value-based 알고리즘인 DQN은 무조건 해당 state에서 가장 높은 누적보상값을 가진 action을 선택하기 때문이다.

Fig. 7.은 20 episodes average rewards가 가장 좋은 Actor-Critic 알고리즘을 사용한 실험의 결과 중 episode에 따른 선택된 action의 수를 나타낸 그래프이다. 현 environment에 가장 적합한 spoofing이 많이 선택되어지고, 나머지 action들은 점차 선택되지 않음을 알 수 있다. Actor-Critic 알고리즘은 가장 높은 누적보상을 주는 action의 선택확률을 높이는 쪽으로 학습이 진행되기 때문에, 가장 누적보상이 큰 spoofing의 선택확률이 가장 커, 상대적으로 선택될 확률이 작은 다른 action들은 선택된 횟수가 적기 때문이다.

네트워크 환경에 있는 PC들의 credential 탈취를 위한 최적의 action들은 학습을 돌릴 때마다 바뀌며, 해당 environment에서는 spoofing이 가장 최적의 action으로 뽑혔다. Actor-Critic은 Q-learning이나 DQN처럼 액션의 선택이 결정론적이지 않기 때문에 episode 초반에서는 많이 튀는 것을 볼 수 있다. 또한 episode 후반에서도 action들의 선택이 현 environment에서 최적의 action인 spoofing으로 완전히 치중되지 않는 모습을 볼 수 있다.

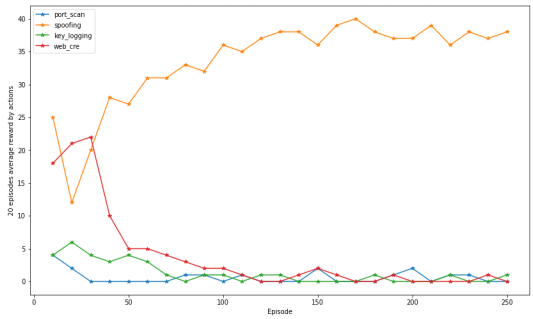

Fig. 7. 20 episodes average reward by actions

Table 5. table of result

Result	Best
Cummulative reward	Actor-Critic
Success rate	Actor-Critic
No. of Iteration	Actor-Critic
The best Action	Spoofing

Table 5.는 지금까지의 실험결과 그래프를 표로 나타낸 것이다. cummulative reward, success rate, No of Iteration 모두 actor-critic 알고리즘이 좋은 성능을 보였고, 해당 환경에서는 spoofing이 가장 효율적인 action이었다.

V. 결 론

본 연구는 실재했던 사이버 공격 시나리오를 가상 환경에서 구현하여 강화학습을 이용해서 해당 네트워크의 취약점이나 효과적인 사이버 공격을 찾고, 자동화된 네트워크 보안을 구성하는 것에 목적이 있다. 기존에 있었던 가상 시나리오 베이스 시뮬레이션 연구와는 달리[8] 본 연구에서는 MITRE ATT&CK의 실제 시나리오를 바탕으로 공격 시뮬레이터를 구현 및 실험하는 데 성공했다. 또한 CyberBattle Sim에 구현되어있는 추상적인 action들이 아닌 실제로 사용되는 해킹 기법을 구현하여 시뮬레이터를 개발하였기 때문에 보다 더 현실성이 있다고 볼 수 있다. 시뮬레이션에 여러 강화학습 알고리즘들을 적용하여 취약점 및 효과적인 공격 방법을 도출하는 것이 가능했다. 이번 시뮬레이션 실험에서는 Actor-Critic 알고리즘이 가장 좋은 결과를 보여주었고, 추후 일반적인 state와 action들을 좀 더 추가하여 특정 시나리오뿐만 아니라 모든 시나리오를 적용시킬 수 있는 프레임워크를 만들 수 있을 것이다.

이번 시뮬레이션 네트워크 환경을 만듦으로써 대규모 네트워크 상황에서는 진행하기 힘든 다양한 공격 모델을 가동시켜볼 수 있고, 사이버 공격 및 수비 시뮬레이션 등을 해볼 수 있을 것이다. 또한 적은 학습 데이터와 많은 변이가 가능한 상황에서도 동작하는 강화학습 기반의 탐지 기술을 확보할 수 있었다.

References

[1] Jeong Do Yoo, Eunji Park, "Cyber Attack and Defense Emulation Agents", Applied Sciences, vol. 10, no. 6, Mar., 2020.

[2] Lee, jung-jai, Jongmin Park, "A Study on Cyber Security Technology", Jounal of The Korea Society of Information Technology Policy & Management (ITPM), pp. 2339-2344, Dec., 2021.

[3] Ho-Gun Rou, Gwang-Yong Gim, "A Study on Detection Method of Web Attack Using Machine Learning", A Study on Detection Method of Web Attack Using Machine Learning, pp. 1642-1650, Jul., 2020.

[4] Sang-Jun Lee, MIN KYUNG IL, "Research on Core Technology for Information Security Based on Artificial Intelligence", The Korea Journal of BigData, pp. 99-108, Dec., 2021.

[5] Chanho Shin, Changhee Choi , "Cyberattack Goal Classification Based on MITRE ATT&CK: CIA Labeling", Journal of Internet Computing and Services, pp. 15-26, Dec., 2022.

[6] Imatitikua D. Aiyannyo, Hamman Samuel, "A Systematic Review of Defensive and Offensive Cybersecurity with Machine Learning", Applied sciences, vol. 10, no. 17, Aug., 2020.

[7] KimJaeuk, Laehyun Park, "Analysis of Deep Learning Model Vulnerability According to Input Mutation", Journal of The Korea Institute of Information Security and Cryptology, pp.51-59, Jan., 2021.

[8] Erich Walter, Kimberly Ferguson-Walter, "Incorporating Deception into CyberBattleSim for Autonomous Defense", arXiv, 2021.

[9] C.P. Andriotis, K.G. Papakonstantinou, "Managing engineering systems with large state and action spaces through deep reinforcement learning", Reliability Engineering & System Safety, Nov., 2019.

[10] Doug Miller, Ron Alford, "Automated Adversary Emulation: A Case for Plannig and Acting with Unknowns", DTIC, Jan., 2018

[11] Erich C. Walter, "Incorporating

deception into cyberbattlesim for autonomous defense", arXiv, 2021
〔12〕 seungeunrho, "Reinforcement Learning from the Floor", youngjin.com, Sep., 2020.
〔13〕 Microsoft 365 Defender Research Team, https://www.microsoft.com/en-us/security/blog/2021/04/08/gamifying-machine-learning-for-stronger-security-and-ai-models/, Apr., 2021.
〔14〕 Mitre attack, "technique", https://attack.mitre.org/, Apr., 2023.
〔15〕 Jungsik Lee, Sung-Young Cho, "A Study on Defense and Attack Model for Cyber Command Control System based Cyber Kill Chain", Journal of Internet Computing and Services, pp. 41-50, Feb., 2021.
〔16〕 Barbara Slavin, Jason Healey, "Iran: How a Third Tier Cyber Power Can Still Threaten the United States", ATLANTIC COUNCIL, Jul., 2013.
〔17〕 Volodymyr Mnih, Koray Kavukcuoglu, David Silver, "Playing Atari with Deep Reinforcement Learning", arXiv, 2013.

〈저자소개〉

김 정 윤 (Jeongyoon Kim) 학생회원
2023년 8월: 서울과학기술대학교 기계시스템디자인공학과 졸업
2023년 9월~현재: 서울과학기술대학교 인공지능응용학과 석사과정
〈관심분야〉 강화학습, 정보보안

박 종 열 (Jongyoul Park) 종신회원
1999년 2월: 광주과학기술원 정보통신공학과 석사
2004년 8월: 광주과학기술원 정보통신공학과 박사
2020년 9월~현재: 서울과학기술대학교 인공지능응용학과 부교수
〈관심분야〉 시각지능, 인지지능, 분산학습

오 상 호 (Sang Ho Oh) 정회원
2016년 8월: 연세대학교 전기전자공학과 석사
2021년 8월: 연세대학교 정보산업공학과 박사
2023년 9월~현재: 국립부경대학교 컴퓨터·인공지능공학부 조교수
〈관심분야〉 정보보호, 인공지능, 머신러닝