*Journal of The Korea Institute of Information Security & Cryptology* ISSN 1598-3986(Print)
VOL.33, NO.6, Dec. 2023 ISSN 2288-2715(Online)
https://doi.org/10.13089/JKIISC.2023.33.6.1021

# 방어 자산의 가용성 상태를 활용한 강화학습 기반 APT 공격 대응 기법*

김 형 록,[1†] 최 창 희[2‡]
[1,2]국방과학연구소 (현역연구원, 선임연구원)

# Reinforcement Learning-Based APT Attack Response Technique Utilizing the Availability Status of Assets*

Hyoung Rok Kim,[1†] Changhee Choi[2‡]
[1,2]Agency for Defense Development (Researcher, Senior Researcher)

## 요 약

국가 지원 사이버 공격은 사전에 계획된 목표를 달성하기 위하여 수행되기 때문에 그 파급력이 크다. 방어자 입장에서 이에 대응을 해야하지만 공격의 규모가 크고 알려지지 않은 취약점이 활용될 가능성도 있기 때문에 대응하기 어렵다. 또한 너무 과한 대응은 사용자의 업무의 가용성을 떨어뜨려서 업무에 지장이 생길 수 있다. 따라서 사용자의 가용성을 확보하면서도 효율적으로 공격을 방어할 수 있는 대응 정책이 필요하다. 본 논문에서는 이를 해결하기 위하여 실시간으로 방어 자산의 프로세스 수와 세션 수를 수집하여 학습에 활용하는 방법을 제안한다. 해당 방법을 활용하여 사이버 공격 시뮬레이터 상에서 강화학습 기반 정책을 학습한 결과, 두 가지 공격자 모델에 대하여 100 time-steps 기준 공격 지속 시간은 각 27.9 time-steps, 3.1 time-steps만큼 감소시켰으며 또한 방어 과정에서 사용자의 가용성을 저해시키는 "복원"행위의 횟수도 감소하여 종합적으로 더 좋은 성능의 정책을 도출할 수 있었다.

## ABSTRACT

State-sponsored cyber attacks are highly impactful because they are carried out to achieve pre-planned goals. As a defender, it is difficult to respond to them because of the large scale of the attack and the possibility that unknown vulnerabilities may be exploited. In addition, overreacting can reduce the availability of users and cause business disruption. Therefore, there is a need for a response policy that can effectively defend against attacks while ensuring user availability. To solve this problem, this paper proposes a method to collect the number of processes and sessions of defense assets in real time and use them for learning. Using this method to learn reinforcement learning-based policies on a cyber attack simulator, the attack duration based on 100 time-steps was reduced by 27.9 time-steps and 3.1 time-steps for two attacker models, respectively, and the number of "restore" actions that impede user availability during the defense process was also reduced, resulting in an overall better policy.

**Keywords:** Machine Learning, Reinforcement Learning, MITRE ATT&CK, CTI, Cyber Simulator

Received(10. 19. 2023), Modified(12. 04. 2023), Accepted(12. 05. 2023)
* 이 논문은 정부의 재원으로 국방과학연구소의 지원을 받아 수행된 연구임.
† 주저자, bbokka123@add.re.kr
‡ 교신저자, changhee84@add.re.kr(Corresponding author)

## I. 서 론

APT(Advanced Persistent Threat) 공격[1]은 구체적인 공격 대상을 향한 악의적인 목적의 지능적이고 지속적인 사이버 공격을 통칭한다. 기존의 단편적인 공격과는 다르게 다양한 공격 기법 및 새롭게 발견된 취약점을 활용하기 때문에 방화벽, 백신, 웹 관제 등과 같이 분야별로 나뉘어져 있는 단편적인 보안으로는 APT 공격을 탐지 및 차단하기 어렵다는 특징이 있다.

APT 공격이 등장한 이후 해당 공격에 사용되는 절차들을 정리하는 다양한 시도[2,3]가 있었는데 MITRE의 ATT&CK 프레임워크[3]가 현재 가장 범용적으로 사용되고 있다. ATT&CK 프레임워크는 실제 공격에 사용되는 공격기법들을 공격방법과 기술의 관점에서 분석하여 TTP(Tactics,Techniques, and Procedures)라는 데이터로 목록화하였다. TTP는 실제 공격 사례를 바탕으로 주기적으로 업데이트되어지며 공격 대상에 따라 Enterprise, Mobile, ICS으로 나뉘어진다.

이렇듯 TTP를 적절히 구현함으로서 다양한 APT 시나리오들을 공격자 입장에서 시뮬레이션 할 수 있고 추가적으로 적절한 방어 행위를 구현하여 공격자와 방어자 간의 행위에 따른 자동화된 공방이 가능하다. 시뮬레이션을 통해 정책을 학습하는 것은 대표적인 방법론에는 강화학습이 있으며 최근에는 고성능의 GPU를 바탕으로 상태의 개수가 많은 환경에서도 좋은 성능을 내는 다양한 심층 강화학습 알고리즘이 등장하였다[4,5,6]. 이에 따라 심층 강화학습과 사이버 공방 시뮬레이터를 활용하여 APT 공격에 대응하는 방어 정책 학습이 가능해졌다.

본 논문에서는 사용자 가용성 측면과 공격 성공률 측면에서 더 나은 성능을 보이는 방어 정책을 학습하기 위하여 내부 자산의 가용성 상태를 모니터링하는 에이전트를 제안한다. 오픈소스 시뮬레이터에서 가장 좋은 성능을 보인 강화학습 알고리즘[6]에 대하여 방어자의 관측값에 자산의 가용성 상태를 추가하여 학습을 시킨 결과 가용성을 모니터링했을 경우 그렇지 않은 경우에 비해 공격이 지속된 시간과 사용자 편의성을 떨어뜨리는 방어자 행위가 모두 감소하였음을 확인할 수 있었다.

## II. 관련 연구

### 2.1 시그니처 기반 APT 공격 대응

초기의 APT 공격 대응[7,8]은 공격의 탐지 이후에 이루어졌다. 방어자가 내부 자산에서 탐지한 악성코드가 기존 APT 공격에 사용되었던 이력이 있을 경우, 이에 대한 매뉴얼적인 대응이 이루어지는 방식이다. FireEye, Mandiant와 같은 보안업체들에서는 위협 인텔리전스 보고서[9,10]를 발행하여 특정 악성코드들의 시그니처 및 기능을 공유한다. 방어자는 이에 대한 백신의 패턴업데이트 및 악성코드의 기능을 확인하여 피해를 인지하고 대응을 하게 된다. 백신 뿐만 아니라 방어자는 IDS, 방화벽과 같이 패킷에 대한 모니터링[11] 또한 수행하며 시그니처 기반으로 공격을 탐지하기 위해 노력해왔다. 하지만 위협 인텔리전스 보고서와 같이 전문가 분석 보고서에 의존하는 사후 분석은 실시간이 아니기에 근본적으로 방어자의 대응은 늦어질 수 밖에 없다.

FireEye 보고서[10]에 따르면 2020년 기준으로 공격자는 방어자 환경에 평균 24일 동안 탐지되지 않은 상태로 있다고 한다. 또한 공격자들은 악성코드 및 페이로드에 약간의 변조를 가하거나 신규 취약점을 활용하여 룰 기반 탐지를 우회하려는 시도를 계속해서 하고 있기에 탐지 자체도 쉽지 않다. 방어자는 기계 학습 등을 활용하여 유사한 악성코드끼리 분류하는 연구[12], 특정 파일이 악성인지 판단하는 연구[13]가 진행되고 있지만 신규 취약점 등을 활용되면 완벽한 탐지가 불가능하다는 점, 실시간적인 대응을 할 수 없다는 점[10]에서 그 한계가 있다.

### 2.2 사이버 공격 대응을 위한 시뮬레이터

위와 같은 시그니처 기반 APT 공격 대응의 한계로 인하여 사이버 공격 환경을 모사하여 실시간적인 공방이 가능한 환경을 통해 공격 및 방어자의 대응 행위를 학습시키는 연구들이 이어졌다. 이러한 연구는 실시간으로 대응이 불가한 시그니처 기반 APT 공격 대응이 가지는 한계를 극복한다.

사이버 공격 환경 모사는 크게 가상의 시뮬레이터를 구현하는 방식과 물리적인 네트워크를 구성하여 실제 환경을 모사하는 에뮬레이터 방식[21]으로 나뉜다. M.E. Kuhl 등[14]은 가상의 공격을 시뮬레이션하기 위하여 네트워크와 IDS를 모델링하였으며

IDS 경보를 통해 보안 시스템의 성능을 측정했다. 강화학습을 최초로 사이버 공격 시뮬레이터에 적용한 Elderman 등[15]은 사이버 공격을 순차적인 의사결정 문제로 단순화하여 접근하고 공격 및 방어 모델을 학습시켰으며, Hammer[16]는 이에 심층 강화학습 모델을 적용하여 성능을 높였다. 위의 두 연구는 사이버 환경을 단순하게 모델링하였다면 이 후, 다양한 국가 기관 및 기업에서 현실의 사이버 공격 환경과 가깝게 모사하기 위한 시도가 이어졌으며 대표적으로 Microsoft 사의 CyberBattleSim[17], MITRE의 FARLAND[18], Australia Defense Science and Technology의 Cyborg[19]가 있다.

본 논문에서는 이러한 공방 환경 중 방어자 정책 학습에 초점을 맞춘 Cyborg 시뮬레이터를 활용하였다. 다른 공방 환경과 달리 Cyborg 시뮬레이터는 여러 강화학습 알고리즘 간의 순위 비교를 공개하여, 제안하고자 하는 방어 자산의 가용성 모니터링의 효과를 정량적으로 분석 가능하다. 본 논문의 실험에서는 Cyborg 시뮬레이터에서 가장 좋은 성능을 보인 알고리즘을 사용하여 방어자의 관측값에 자산의 가용성 상태를 추가하였을 때 사용자의 가용성을 확보하면서도 방어 성공률 측면에서 더 나은 대응 정책을 학습 가능하다 것을 보인다.

### 2.3 강화학습 방법론

강화학습은 기계학습의 한 영역으로 어떤 환경 안에서 정의된 에이전트가 현재의 상태를 인식하여, 선택 가능한 행동들 중 보상을 최대화하는 행동 혹은 행동 순서를 선택하는 방법을 학습한다. 초기에 강화학습에서 다루던 환경은 주로 마르코프 결정 프로세스로서 주어졌으며 Dynamic Programming와 같은 기법을 이용하여 해결할 수 있었지만 실제 환경에서 전이 확률 행렬 및 보상함수에 대한 정보를 얻기 힘들기 때문에 현실의 문제는 마르코프 결정 프로세스에 대한 모든 정보를 알기 어렵다.

사이버 공격 시뮬레이터도 마찬가지로 다양한 종류의 사용자들이 존재하기 때문에 개별 행위자에 대한 상태 전이 확률 행렬을 구할 수 없다. 이와 같이 마르코프 결정 프로세스의 모든 정보를 알 수 없을 때 모델 프리 알고리즘을 사용하게 되고 모델 프리 알고리즘[4,5,6]에는 가치 기반 에이전트 학습, 정책 기반 에이전트 학습와 가치 함수와 정책 함수를 혼합해서 사용하는 액터-크리틱 방법론이 있다. 특정 방법론이 모든 상황에서 좋은 성능을 내는 것은 아니며 강화학습이 이루어지는 환경마다 적합한 알고리즘이 다르다. 사이버 공격 시뮬레이터 중 하나인 Cyborg[19]는 2021년부터 현재까지 매년 Cage-Challenge[20]를 열어 해당 환경에서 가장 좋은 성능을 내는 알고리즘을 선발하는데 정책 기반 에이전트 중 하나인 PPO (Proximal Policy Optimization) 알고리즘[6]이 가장 좋은 성능을 내는 것을 확인할 수 있다. 본 논문의 실험에서도 제안된 방어자 에이전트를 학습하는데 있어 Cyborg 환경을 활용하였으며 PPO 알고리즘이 해당 환경에서 가장 좋은 성능을 낸다는 점과 time-step 기반으로 보상을 받는 알고리즘 중 확률적 정책을 반환하고 샘플 효율성 측면에서 우수한 PPO알고리즘을 활용하여 실험을 진행하였다.

## III. APT 공격 시뮬레이터

본 절에서는 제안된 방어자 에이전트를 학습 및 테스트하기 위한 오픈소스 공격 시뮬레이터의 구성에 대한 설명을 포함한다. 사이버 공격과 방어가 이루어지는 환경은 Cage - Challenge2[20]의 테스트 베드를 사용하였다.

### 3.1 테스트 베드 구성

테스트 베드의 네트워크는 그림 1과 같이 3개의 서브넷으로 나뉘어져 있다. 가장 바깥쪽 서브넷은 외부와 연결된 영역으로 사용자 영역만 존재하며 이를 서브넷1이라 칭한다. 해당 영역의 사용자들은 내부 라우터를 통하여 서브넷2에 존재하는 각종 내부 엔

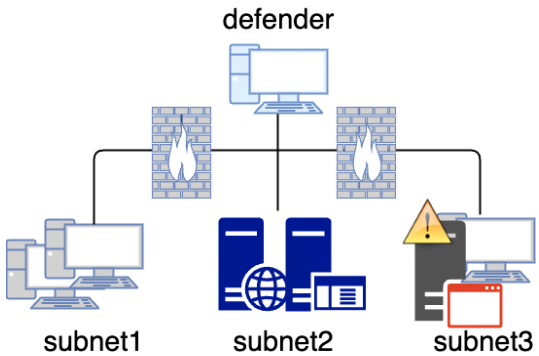



Fig. 1. Testbeds structure as specified in the environment of the CAGE Challenge2 [20].

터프라이즈 서버 및 홈페이지에 접근할 수 있다. 방어자 호스트는 전체 자산과 세션을 맺기 위하여 중간에 위치한 서브넷2에 존재한다. 실제 공격자가 목표로 하는 자산은 서브넷3에 존재하며 서브넷3에는 공격하고자 하는 서버 외에 같은 서브넷에 위치하는 내부망 사용자 PC들이 존재한다.

### 3.2 APT 공격행위자 및 정상행위자 모델링

본 시뮬레이터에서는 공격자가 방어자 네트워크의 가장 외부에 위치해 있는 서브넷1의 어느 한 유저 호스트의 관리자 권한 세션을 피싱을 통하여 획득하였다는 것을 가정하고 시작한다. 이 후 공격자의 목적은 해당 거점 PC로부터 순차적으로 서브넷2를 장악한 뒤 서브넷3에 위치해 있는 내부 서버에서 구동되고 있는 서비스를 중단시키는 것이다. 해당 서비스를 이후 "코어 서비스"라고 표현한다.

#### 3.2.1 공격 행위자의 행위

해당 시뮬레이터에서 공격자의 행위는 크게 다섯 가지로 구성되고 각 행위는 MITRE ATT&CK의 technique과 대응된다.

첫 번째 행위는 "네트워크 스캐닝(T1018)"에 해당한다. 해당 행위를 통하여 공격자는 같은 대역에 존재하는 IP들의 목록을 획득한다. 두 번째 행위는 "원격 서비스 목록 획득(T1046)"으로 같은 대역의 특정 호스트에 존재하는 원격 서비스들의 목록을 획득한다. 공격자는 해당 행위를 통하여 취약한 원격 서비스를 포함한 원격 서비스 목록 및 버전, 포트에 대한 정보를 얻게 된다. 세 번째 행위는 "원격 서비스 익스플로잇(T1210)"으로 T1046으로 획득한 원격 서비스 목록 중 취약한 원격 서비스들 중 하나를 선택하여 익스플로잇을 하게 된다. 해당 행위가 성공하게 되면 공격자는 대상 호스트의 USER 권한 세션을 획득하게 된다. 네 번째 행위는 "권한 상승(TA0004)"으로 이전 단계에서 획득한 USER 권한 세션을 권한 상승 취약점을 활용해 관리자 권한으로 변경시킨다. 해당 공격이 성공하게 되면 공격자는 해당 PC의 모든 Interface정보를 획득하여 두 개의 서브넷에 소속돼있는 호스트의 경우 내부 서브넷에 대해서 "네트워크 스캐닝"과 같은 행위를 수행할 수 있다. 마지막 행위는 "서비스 중지(T1489)"로 공격자가 목표로 하는 서브넷3의 내부 서버에 접근하게 되면 해당 행위를 통하여 코어 서비스를 중지시키게 된다. 공격자는 한번의 에피소드 내에서 코어 서비스 중지 시간을 최대한 길게하는 것이 목적이다.

#### 3.2.2 정상 행위자의 행위

일반적인 내부망 환경에서는 방어자와 공격자뿐 아니라 정상적인 행위를 수행하는 일반 사용자들이 존재한다. 해당 사용자들이 환경에 미치는 영향이 상당히 크기 때문에 시뮬레이터에서 일반 사용자들이 자산의 상태에 영향을 미치는 행위 두 가지를 선정하였다.

첫 번째는 사용자가 서버에 돌아가고 있는 서비스를 이용하는 것이다. 시뮬레이터 상으로 정상 사용자와 서버의 프로세스 간에 세션이 업데이트 되게 된다. 두 번째 행위는 특정 서브넷에 ping을 보내는 행위로서 이는 공격자의 T1018을 수행하는 행위와 동일한 결과를 나타낸다. 다만 정상사용자가 접근할 수 있는 대역의 초기값은 서브넷1과 서브넷2로 설정한다.

## IV. 방어자산의 가용성 상태 기반 강화학습 정책 학습

본 논문에서는 방어 자산의 가용성 상태를 모니터링하여 강화학습의 입력 벡터로서 활용함으로서 APT 공격 대응 정책의 성능 향상을 목표로 한다. 방어 행위자는 그림 2와 같이 본인이 모니터링하고 있는 관측값과 이전 행위의 결과를 바탕으로 정책을 학습하게 된다. 정책을 학습하는 자세한 과정은 4.4.절에서 다룬다. 정상 행위자와 공격 행위자의 정책은 학습시키지 않으며 3.2.절의 행위를 초기 프로그래밍된대로 행동하게 된다. 정상 행위자는 3.2.절에서 서술한 행위를 랜덤으로 선택하여 실행한다. 공격 행위자는 서브넷3의 공격 목표 서버에 도달할 때까지 네트워크 스캐닝, 원격 서비스 목록 획득, 익스플로잇, 권한 상승의 순으로 행위를 선택하며 최종적으로 공격 목표 서버에 서비스 중지 행위를 수행한다. 이전 단계의 공격이 방어자의 행위로 인하여 실패하였다면 해당 공격부터 진행한다.

### 4.1 방어 자산의 가용성 상태

방어자산의 가용성 상태란 각 방어 자산에 실행되

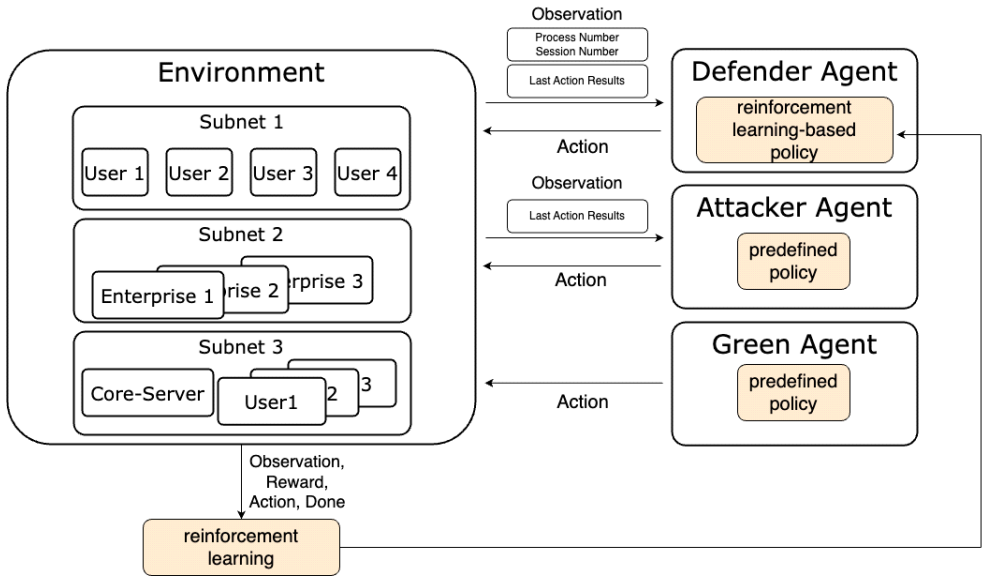



Fig. 2. Diagram illustrating the observations and actions of each participant in the cyberattack simulator. Action proceeds in follow order: defender agent, green agent, attacker agent. The defender selects actions based on the learned algorithm, while the attacker agent and the green agent act according to predefined policies.

고 있는 프로세스의 개수와 세션의 개수로서 구성되어진다. 방어자는 공격 시나리오가 진행됨에 있어 변화되는 각 자산의 가용성 상태 값을 입력으로 받아 자신의 다음 행위를 결정한다. 프로세스 개수와 세션의 개수는 거시적인 방어 자산의 상태를 나타내는 주요한 값일 뿐 아니라 방어자 입장에서 현실적으로 모니터링이 가능한 값으로서 실제 환경에서 정책 학습 시에 활용이 가능하다는 장점이 있다.

방어 자산의 가용성 상태는 그림 3과 같이 정상 행위자 및 공격 행위자의 행위로 인해 수시로 변경된다. 그림 3의 Blue Agent, Red Agent, Green Agent는 각각 방어 행위자, 공격 행위자, 정상 행위자를 의미한다. 공격자의 익스플로잇으로 인해 실행된 악성코드는 신규 프로세스와 세션을 생성한다. 또한 방어자가 악성코드 제거 및 "복원"을 통한 공격자 세션 제거 등과 같은 적절한 행위를 취했다면 방어 자산의 가용성 상태 마찬가지로 변화하게 된다. 가용성 상태는 실제 벡터화 시, 각 호스트에 존재하는 프로세스의 개수, 세션의 개수로서 환경으로부터 전달이 되기 때문에 총 13개의 호스트에 대하여 크기 26의 벡터로서 표현이 된다. 각 호스트에 실행되고 있는 프로세스 개수와 세션 개수를 각각 최대 프로세스 개수 및 최대 세션 개수로 나눈 값을 입력 벡터로 사용하였다.

### 4.2 방어 행위자 행위

방어 행위자는 테스트 베드에 존재하는 모든 자산과 세션을 맺은 상태로서 각 자산에 명령을 내려 가용성 상태를 모니터링한다. 테스트 베드 상 방어 행위자 호스트는 서브넷2에 위치하여 서브넷1,3 모두와 직접적인 통신이 가능하다. 방어자산의 가용성에 대한 모니터링 행위는 매 time-step마다 진행되고 이 외에 공격자의 침해로부터 대응하기 위해 크게 네 가지 행위를 시뮬레이션하였다.

첫 번째 행위는 "분석"으로 특정 호스트를 대상으로 해당 호스트 내의 파일을 분석한다. 백신 정밀 검사를 하는 행위에 대응되며 각 파일의 악성 판단 정도를 반환한다. 시뮬레이터 상으로는 악성 파일의 경우 해당 값을 높게 설정하였다. 두 번째 행위는 "유인 서비스 등록" 행위로서 특정 호스트에 정상 사용자가 사용하지 않는 서비스를 등록한다. 이를 통해 공격자의 공격을 지연시킬 수 있다. 세 번째 행위는 "파일 제거"로 "분석" 행위를 통해 획득한 악성 의심 파일을 제거한다. 마지막 행위는 "복원"으로 복원의 행위를 거치게 되면 특정 호스트에 존재하는 공격자

| Host | User1 | User2 | Enterprise1 | Enterprise2 | … |
|---|---|---|---|---|---|
| Process Count | 13 | 8 | 5 | 7 | … |
| Session Count | 3 | 4 | 3 | 3 | … |

| Blue Agent | Monitoring Availability |
|---|---|
| Red Agent | Exploit User2 |
| Green Agent | Use Enterprise1 Service |

| Host | User1 | User2 | Enterprise1 | Enterprise2 | … |
|---|---|---|---|---|---|
| Process Count | 13 | 9 | 6 | 7 | … |
| Session Count | 3 | 5 | 3 | 3 | … |

Fig. 3. Examples of availability state changes due to the actions of three types of actors

의 세션이 사라지게 된다. 만약 특정 호스트가 장악당하여 공격자의 관리자 권한 세션이 존재한 상태에서는 오직 "복원" 행위를 통해서만 공격자의 세션을 끊을 수 있다. 이 행위를 통해 공격자는 내부 자산 중 하나의 거점을 잃게 된다. 방어 자산의 가용성 상태 모니터링 외의 행위들은 오픈소스 시뮬레이터[19]를 활용하여 구현하였으며 공격자가 어느 원격 서비스를 "익스플로잇"할지는 랜덤으로 결정하기 때문에 방어자의 행위가 모든 공격을 막을 수는 없게 설정하였다.

### 4.3 학습 알고리즘

시뮬레이터 내에서 행위를 선택하고 실행하는 순서는 방어자, 정상행위자, 공격행위자 순으로 설정한다. 방어 행위자 외의 행위자들의 정책은 학습을 시키지 않고 방어 행위자의 정책은 충분히 많은 수의 경험을 쌓고 업데이트 시키기 때문에 시뮬레이터 내 행위자의 수행 순서는 크게 영향을 미치지 않는다.

그림 2와 같이 방어자가 환경으로부터 가용성 상태 및 그전 행위에 대한 결과값을 받게 되면 학습된 정책을 바탕으로 행위를 선택 및 실행한다. 순차적으로 정상행위자, 공격 행위자의 행위가 이루어진다. 이 과정에서 변화되는 환경에 대한 상태 값과 그에 따른 행동 및 보상 값을 저장하여 방어자 정책 학습에 사용하게 된다. 해당 시뮬레이터에서는 보상을 time-step 기준으로 받도록 설정하였고 20000번의 time-steps만큼 경험을 쌓고 정책을 업데이트하였다. 2.3.절에서 소개한 바와 같이 PPO 알고리즘을 강화학습 알고리즘으로 사용하였다. 사용된 알고리즘의 loss function.은 수식(1)과 같다. $r(\theta)$는 전 정책에서의 확률에 대한 현 정책의 확률을 의미하고 A는 상태에 대한 가치 추정값과 보상의 기대값과의 차이를 의미한다.

$$
\begin{aligned}
L(\theta) &= E[\min(r(\theta)*A, clip(r(\theta), \\
&\qquad (1-\epsilon, 1+\epsilon)*A)] \\
L_{critic}(\theta_v) &= E[(V_{\theta_v} - V_{target})^2] \\
L_{total}(\theta, \theta_v) &= L(\theta) + L_{critic}(\theta_v)
\end{aligned}
\tag{1}
$$

### 4.4 강화학습 보상 함수

해당 강화학습 알고리즘은 100 time-steps 기준으로 하나의 episode가 구성되며 episode 기준으로 보상 리스트가 초기화된다. 하나의 time-step이 끝날 때마다 방어 자산의 특징 값을 측정하여 보상이 주어진다. 해당 특징값에는 코어 서비스 동작 여부, 자산 내 공격 세션의 수, "복원" 작업 수행 여부가 있다. 공격자 에이전트가 방어자산에 관리자 권한의 세션을 형성할 때마다 음의 보상을 얻게 되고 기존에 형성된 공격자 세션에 대하여서도 세션이 사라지지 않는 이상 지속적으로 음의 보상을 얻게 된다. 또한 공격자가 최종 공격 목표, 즉, 서브넷3에 위치한 내

부 서버의 코어 서비스를 중지시켰을 경우에 음의 보상을 얻게 된다. 또한 방어자 행위 중 하나인 “복원” 작업을 수행했을 경우 음의 보상을 얻는다. 수식(2)는 t번째 time-step의 보상 값으로서, 각각의 하이퍼 파라미터 값을 달리해가며 최적의 보상함수 가중치를 결정하게 된다

$$R(s,a) = -\begin{Bmatrix} \alpha^*(\text{코어 서비스 동작 여부}) \\ +\beta^*(\text{자산 내 공격 세션 수}) \\ +\gamma^*(\text{복원 작업 수행 여부}) \end{Bmatrix} \quad (2)$$

최적 정책의 목적은 코어서비스 동작을 최대한 보장시키면서 방어 행위자의 "복원“ 행위를 최소한으로 하는 것이다. 무분별한 "복원“ 행위는 정상적인 사용자 업무의 방해를 유발할 수 있다. 특히 각 자산마다 "복원“이라는 행위가 가지는 편의성 저하 정도가 다르기 때문에 사용자 호스트와 서버에 대한 가중치를 다르게 하여 서버에 수행하는 "복원“ 비용이 더 크도록 설정하였다.

### 4.5 강화학습 정책 출력값

강화학습 모델에 방어자산의 가용성 상태 값 및 그 전 행위에 대한 관측값을 입력값으로 넣게 되면 결과값으로 다섯가지의 방어자 행위와 해당 행위를 실행할 호스트의 주소를 인덱싱한 결과가 반환된다. 기본 모니터링 행위의 경우 특정 호스트가 아닌 시스템 전체에 대한 가용성 상태를 모니터링하는 것으로 하나의 행위로 표현되고 나머지 행위들은 총 13개의 자산이 존재하기 때문에 각 13가지의 행위가 존재하게 된다. 가용성 모니터링 행위는 모든 자산에 대해 매 time-step 실행되며 이 외의 방어자 행위는 한 time-step에 하나의 자산에만 실행할 수 있다.

## V. 실험 결과

본 실험에서는 두 가지 APT 공격자 모델에 대해 학습 및 테스트를 수행하였다.

첫 번째 공격자는 방어자 자산의 망 구조를 알고 최적의 경로로 공격을 하는 공격자이다. 이러한 공격자를 "최적 경로 공격자“라고 칭한다. 해당 공격자는 현재 위치보다 내부에 위치해 있는 서브넷으로 이동할 때마다 장악해야하는 호스트를 명확히 알고 해당 호스트들에 대한 공격을 하는 공격자이다. 즉, 사전의 정의된 최적화된 공격루트를 통해 서브넷3의 내부 서버에 침투한다. 두 번째 공격자는 망의 구조를 모르는 공격자로서 차례대로 같은 서브넷에 있는 취약한 방어 자산들의 세션을 획득하고 다른 서브넷으로의 이동이 가능해지면 서브넷을 이동하는 식으로 최종 목표 자산까지 접근하는 "순차적 경로 공격자“이다. 학습 시에는 "최적 경로 공격자"를 대상으로 방어자 정책을 학습하며 학습의 추이에 대해 분석한다.

이 후, 가용성 모니터링 전과 후의 성능을 비교하고 추가적으로 “순차적 경로 공격자"를 대상으로 테스트를 한다. 5.1절에서는 식(2)의 보상함수의 각 요소에 대한 가중치를 변경해가며 코어 서비스 동작 시간과 복원 행위 횟수의 변화를 분석하여 가장 좋은 성능을 내는 보상함수 가중치를 찾아, 5.2.절에서 해당 보상함수에서 방어자산의 가용성 모니터링이 가져다 주는 성능 변화에 대해 다룬다.

### 5.1 보상 함수 가중치 변경에 따른 성능 분석

학습된 모델의 성능 지표는 식 (3)으로 표현되며 “impacted duration"은 하나의 에피소드 내에서 코어 서비스가 중지된 시간, “restore count"는 "복원“ 행위 수행 횟수를 의미한다.

$$score = 100/(impacted\,duration + restore\,count) \quad (3)$$

학습은 "최적 경로 공격자"를 대상으로 10000번의 에피소드로 이루어지며 하나의 에피소드는 100 time-steps으로 설정한다. PPO모델의 업데이트 주기는 20000 time-steps이며 하이퍼 파라미터는 각각 learning rate=0.002, discount factor=0.99, clipping parameter=0.2, epoch=6으로 설정하였다. 보상함수 가중치의 경우 코어 서비스 동작 여부의 가중치를 10으로 고정시키고 나머지 가중치를 수정하며 방어자 성능을 측정하였다. “복원” 행위 가중치는 자산이 사용자 PC인지, 서버인지, 공격 목표 서버인지에 따라 중요도를 1:2:3으로 부여하였다. 실제 가중치는 (1,2,3), (2,4,6), (3,6,9) 세 개로 나누어 실험을 진행하였다. 공격자 세션 개수에 대한 가중치는 대상이 호스트와 서버로 나누어 중요도를 1:10으로 부여하였다. 실제 가중치는 (0.1,1), (0.2,2), (0.3,3)으로 나누어 실험하였다.

총 9가지 조합의 가중치에 대해 10000번의 에피

Table 1. Performance analysis for 9 different weighting combinations against an "optimal-path attacker"

| "Restore" weight / Attacker session weight | (1, 2, 3) | | (2, 4, 6) | | (3, 6, 9) | |
|---|---|---|---|---|---|---|
| | Impacted Duration | Restore Count | Impacted Duration | Restore Count | Impacted Duration | Restore Count |
| (0.1, 1) | 19.32 | 6.28 | 41.1 | 4.52 | 27.58 | 2.68 |
| (0.2, 2) | 23.1 | 12.62 | 18.6 | 4.3 | 37.24 | 7.14 |
| (0.3, 3) | 12.74 | 73.92 | 34.6 | 12.32 | 52.86 | 9.04 |

소드만큼 학습하였으며 마지막 50 에피소드의 평균 성능은 위 표 1에 표시한 바와 같다. "복원" 행위 비용에 대한 가중치를 높일수록 "복원" 행위 수행 횟수 자체는 감소하는 추세를 보였다. 하지만 상대적으로 코어 서비스 중지에 대한 가중치가 줄어드는 효과를 보여 (3,6,9)로 설정하였을 때 (1,2,3)의 가중치를 두었을 때보다 모든 경우에서 서비스 중지 시간이 증가하였다. 또한 자산 내 공격자 세션 개수에 대한 가중치를 증가시켰을 때 "복원" 행위가 증가함을 확인할 수 있다. 이는 자산 내부 공격자의 세션을 소멸시키기 위해 "복원" 행위가 상대적으로 자주 수행되기 때문이다. 해당 가중치의 크기에 따라 서비스 중지 시간은 단축되지만 "복원" 행위가 너무 빈번하게 일어나 실제로 활용하기 힘들다는 단점이 있다. 총 9가지 경우 중 코어 서비스 중지 시간과 "복원" 행위의 수행 횟수의 합이 가장 작은 경우는 각 가중치 값의 중간 값을 선택한 경우로서 공격자 세션에 대한 가중치를 (0.2,2)로 두고 "복원" 행위에 대한 가중치를 (2,4,6)으로 두었을 때이다.

보상 함수를 위의 가중치대로 설정하고 방어자 정책을 학습 시켰을 때의 결과는 그림 4의 파란색 선과 같다. 그림 4(b)의 파란색 선은 코어 서비스 중단 시간을 나타내고 처음 50 에피소드에서 평균 75.98 time-steps에서 시작하여 마지막 50 에피소드에서는 평균 18.6 time-steps으로 점차적으로 감소함을 확인할 수 있다. 동시에 "복원" 행위 또한 그림 4(c)의 파란색 선에서 확인할 수 있듯이 8.86회에서 4.3회로 감소하여 실제 방어자가 의도한대로 "복원"를 제외한 다른 행위를 활용하여 공격에 대응하는 식으로 정책 학습이 올바르게 진행되고 있음을 알 수 있다.

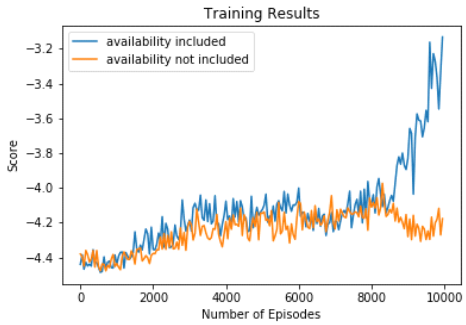
### 5.2 가용성 모니터링에 따른 성능 변화

5.1절에서 가장 좋은 성능을 냈던 보상함수에 대해 가용성 모니터링을 한 강화학습 모델과 하지 않은 학습 모델에 대한 성능 비교 결과는 그림 4(a)와 같다. 그림 4(b)를 보면 코어 서비스 중지 시간의 경우 가용성 모니터링 결과가 포함되었을 경우 10000번 학습 기준 18.6 time-steps까지 떨어졌지만 포함되지 않았을 경우는 46.5 time-steps에서 머물렀다. 또한, 그림 4(c)를 확인하여 보면 "복원" 횟수의 경우에도 가용성 모니터링을 한 학습 모델이 그렇지 않은 모델에 비해 14.4회 더 적게 발생하여 종합적으로 더 좋은 성능을 보였다. 이는 정책 학습 시에 각 자산에 존재하는 프로세스 수와 세션 수를 유의미한 특징 값으로 사용하고 있음을 의미한다.

"최적 경로 공격자"를 대상으로 학습시킨 에이전트를 "순차적 경로 공격자" 대상으로 테스트한 결과는 표 2와 같다. 한 에피소드의 크기를 30,50,100 time-steps으로 세 가지 경우에 대하여 성능을 평가하였다. 테스트는 각각의 경우에 대해 1000번의 에피소드만큼 진행하였으며 표2의 수치는 그 평균이

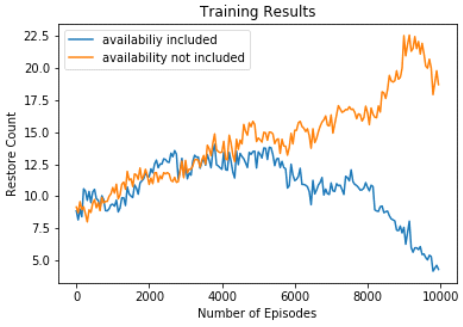
Table 2. Test results for "Sequential Path Attacker"

| step size | Impacted Duration | | Restore Count | |
|---|---|---|---|---|
| | Availability Considered | Not Considered | Availability Considered | Not Considered |
| 30 | 0.02 | 0.12 | 1.34 | 7 |
| 50 | 1.4 | 2.74 | 1.9 | 11.46 |
| 100 | 15.08 | 18.18 | 3.98 | 23.58 |




(a) Score for "Optimal Attacker"


(b) Impacted Duration for "Optimal Attacker"




(c)Restore Count for "Optimal Attacker"

Fig. 4. Results comparing with and without availability information for 10000 training episodes

다. 모든 경우에서 가용성을 고려한 정책이 고려하지 않은 정책보다 더 높은 성능을 보였다. 100 time-steps으로 구성된 에피소드의 경우에는 평균 3.1 time-steps만큼 서비스가 중지 시간을 감소시켰고 그 과정에서 수행된 "복원" 행위는 평균 19.6회 감소되었다. 가용성 모니터링이 추가된 강화학습 정책이 "최적 경로 공격자"뿐만 아니라 "순차적 경로 공격자'에 대해서도 더 좋은 성능을 나타낸 것은 해당 특징 값이 보편적으로 강화학습 정책에 있어서 주요한 특징 값으로서 작용한다는 것을 의미한다.

## VI. 결 론

논문에서는 자산의 가용성 상태를 관측값으로 한 강화학습 기반 방어자 정책의 유효성을 사이버 공격 시뮬레이터를 활용하여 검증하였다. 방어 자산의 가용성 상태는 현실적으로 모니터링이 가능한 값이라는 점에서 그 의의가 있다. 100 time-steps 시뮬레이션 기준으로 실험 결과에 따르면 가용성을 고려한 정책은 고려하지 않은 정책에 비해 "순차적 경로 공격자" 대상은 3.1 time-steps, "최적 경로 공격자" 대상은 27.9 time-steps만큼 코어 서비스 중지 시간을 감소시켰다. 방어 과정에서 사용자의 편의성을 떨어뜨리는 "복원"행위의 횟수 또한 가용성을 고려한 정책에서 그렇지 않은 정책에 비해 더 적게 발생하여 종합적으로도 더 높은 성능을 보였다. 다만, 사이버 공격 시뮬레이터를 활용하여 방어자 행위의 종류를 특정해놓은 상태에서 실험하였기 때문에 실험 결과는 방어 행위의 종류에 따라 변할 수 있다. 따라서 후속 연구로는 더욱 다양한 방어 정책 하에서 일반적으로 좋은 성능을 나타내는 정책을 학습시키는 것을 목표로 할 것이다.

## References


[1] P. Chen, L. Desmet and C. Huygens, "A study on advanced persistent threats," 15th IFIP International Conference on Communications and Multimedia Security (CMS), pp.63-72, Sep. 2014.

[2] LockheedMartin, "The Cyber Kill Chain," https://www.lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html, accessed 2023.10.19.

[3] MITRE, "ATT&CK Framework,"https://attack.mitre.org, accessed 2023.10.19.

[4] V. Mnih, K. Kavukcuoglu and D. Silver, A. Graves, I. Antonoglou, D. Wierstra and M. Riedmiller, "Playing atari with deep reinforcement learning," arXiv preprint arXiv: 1312.5602., Dec. 2013.

[5] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, "Trust

region policy optimization," In Proceedings of The 32nd International Conference on Machine Learning, pp. 1889 - 1897, Jun. 2015.
[6] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," arXiv preprint arXiv:1707.06347, Aug. 2017.
[7] Cho, D. Xuan, and H. H. Nam, "A method of monitoring and detecting APT attacks based on unknown domains," Procedia Computer Science 150, pp.316-323, 2019.
[8] S. M. Milajerdi, R. Gjomemo, B. Eshete, R. Sekar and V. N. Venkatakrishnan, "Holmes: real-time apt detection through correlation of suspicious information flows," In 2019 IEEE Symposium on Security and Privacy (S&P), pp.1137-1152, May. 2019.
[9] Mandiant, "Cyber Espionage is Alive and Well: APT32 and the Threat to Global Corporations," https://www.mandiant.com/resources/blog/cyber-espionage-apt32, accessed 2023.10.19.
[10] FireEye Inc., "M-Trends 2021:Cyber Security Insights: Technical Reports," https://vision.fireeye.com/content/fireeye-vision/en-us/editions/11/11-m-trends.html, accessed 2023.10.19.
[11] Hamed Salehi, Hossein Shirazi and R.A. Moghadam, "Increasing overall network security by integrating signature-based NIDS with packet filtering firewall," In 2009 International Joint Conference on Artificial Intelligence, pp.357-362, Apr. 2009.
[12] C.I. Fan, H.W. Hsiao, C. H. Chou and Tseng, Y. F., "Malware detection systems based on API log data mining," In 2015 IEEE 39th annual computer software and applications conference, vol.3, pp.255-260, Jul. 2015.
[13] F. Karbalaie, A. Sami and M. Ahmadi, "Semantic malware detection by deploying graph mining," International Journal of Computer Science Issues (IJCSI), vol.9, issue 1, no.3, Jan. 2012.
[14] M.E. Kuhl, M. Sudit, J. Kistner and K. Costantini, "Cyber attack modeling and simulation for network security analysis." In 2007 Winter Simulation Conference, pp.1180-1188, Dec. 2007.
[15] R. Elderman, L.J. Pater, A.S. Thie, M.M. Drugan and M.A. Wiering, "Adversarial reinforcement learning in a cyber security simulation," In 9th International Conference on Agents and Artificial Intelligence (ICAART) ,pp. 559-566, Jan. 2017.
[16] K. Hammar and R. Stadler, "Finding effective security strategies through reinforcement learning and self-play," In 2020 16th International Conference on Network and Service Management (CNSM), pp.1-9, Nov. 2020.
[17] Microsoft Threat Intelligence, "Gamifying machine learning for stronger security and AI models," https://www.microsoft.com/en-us/security/blog/2021/04/08/gamifying-machine-learning-for-stronger-security-and-ai-models/, accessed 2023.10.19
[18] A. Molina-Markham, C. Miniter, B. Powell and A. Ridley, "Network environment design for autonomous cyberdefense," arXiv preprint arXiv:2103.07583., Mar. 2021.
[19] C. Baillie, M. Standen, J.Schwartz, M.Docking, D. Bowman and Junae Kim, "Cyborg: An autonomous cyber operations research gym," arXiv preprint arXiv:2002.10667, Feb. 2020.
[20] Github, "Cyber Autonomy Gym for

Experimentation Challenge 2," https://github.com/cage-challenge/cage-challenge-2,accessed 2023.10.19.
[21] Akbari, I., Tahoun, E., Salahuddin, M.A., Limam, N., & Boutaba, R., "ATMoS: Autonomous threat mitigation in SDN using reinforcement learning," In NOMS 2020 IEEE/IFIP NetworkOperations and Management Symposium, pp.1-9, April. 2020.

## 〈저자소개〉

김 형 록 (Hyoung Rok Kim) 정회원
2019년 2월: 고려대학교 사이버국방학과 학사
2019년 8월~2022년 6월: 사이버작전사령부
2022년 7월~현재: 국방과학연구소 국방첨단과학기술연구원 사이버센터 현역연구원
〈관심분야〉 정보보호, AI, 사이버 모의전투

최 창 희 (Changhee Choi) 정회원
2008년 2월: 연세대학교 컴퓨터과학과 학사
2010년 2월: 한국과학기술원 전산학과 석사
2013년 8월: 한국과학기술원 전산학과 박사
2013년 9월~현재: 국방과학연구소 국방첨단과학기술연구원 사이버센터 선임연구원
〈관심분야〉 정보보호, 사이버전, 머신러닝 기반 사이버 보안, AI, GAN