Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.31, NO.5, Oct. 2021 ISSN 2288-2715(Online)
https://doi.org/10.13089/JKIISC.2021.31.5.1045

iRF: 대규모 사이버 방어 훈련을 위한 통합 레드팀 프레임워크

장 인 숙,[1†] 조 은 선[2‡]
[1,2]충남대학교 (대학원생, 교수)

iRF: Integrated Red Team Framework for Large-Scale Cyber Defence Exercise

In Sook Jang,[1†] Eun-Sun Cho[2‡]
[1,2]Chungnam National University (Graduate student, Professor)

요 약

APT 공격이 빈번해지고, 정교해짐에 따라 보안시스템의 고도화뿐만 아니라 이를 운영하는 각 기관의 정보보호 담당자의 역량이 점점 중요해지고 있다. 다수의 블루팀(BT)이 참가하고 기관망 모사 및 방어 대상 시스템이 많은 대규모 훈련 운영 시에는 다양한 공격 패턴, 네트워크 페이로드, 시스템 이벤트를 생성할 수 있도록 공격을 모사할 수 있어야 한다. 그러나 하나의 레드팀(RT) 프레임워크를 사용할 경우 블루팀에 의해 쉽게 탐지될 수 있다는 한계가 있으며 수십 개 이상 다수의 RT 프레임워크를 운영할 때는 각 프레임워크 별로 훈련 설정 및 운영을 위한 전문가의 많은 시간과 노력이 필요하다. 본 논문에서는 다수의 공개용 RT 프레임워크와 직접 제작한 RT 프레임워크 등을 통합하여 대규모 훈련을 자동으로 운영할 수 있는 통합 프레임워크(iRF)를 제안한다.

ABSTRACT

As APT attacks become more frequent and sophisticated, not only the advancement of the security systems but also the competence of the cybersecurity officers of each institution that operates them is becoming increasingly important. In a large-scale cyber defence exercise with many blue teams participating and many systems to simulate and defend against, it should be possible to simulate attacks to generate various attack patterns, network payloads, and system events. However, if one RT framework is used, there is a limitation that it can be easily detected by the blue team. In the case of operating multiple RT frameworks, a lot of time and effort by experts for exercise setup and operation for each framework is required. In this paper, we propose iRF(integrated RT framework) that can automatically operate large-scale cyber defence exercise by integrating a number of open RT frameworks and RT frameworks created by ourselves.

Keywords: Cyber Defense Exercise, Red Team Framework, Cybersecurity Training

I. 서 론

APT 공격이 빈번해지고, 정교해짐에 따라 보안시스템의 고도화뿐만 아니라 이를 운영하는 각 기관의 정보보호 담당자의 역량이 점점 중요해지고 있다[1]. APT 공격에 대한 대응 역량 강화 훈련을 운영하기 위해서는 기관망 내부의 다양한 시스템, 네트워크, 보안시스템 등의 데이터 소스로부터 발생하는 탐지 요소들을 식별하고 수집 및 가시화하여 실제 공격에 해당하는 이벤트를 탐지 및 차단하도록 방어 연습을 수행할 수 있는 여건을 갖추어야 한다. 이에 따라, NATO의 LockedShilelds[12], EU의 Cyber Europe[13], 이스라엘의 CyberGym[14], 스웨덴의 CRATE[15] 등과 같은 국가 연합 및 정부 주도

Received(08. 17. 2021), Modified(09. 13. 2021), Accepted(09. 13. 2021)
† 주저자, isjang2k@gmail.com
‡ 교신저자, eschough@cnu.ac.kr(Corresponding author)

의 사이버 훈련장이 구축 및 운영되고 있으며, 국내에서도 한국인터넷진흥원의 온라인 실전형 사이버 훈련장[16], 사이버안전훈련센터의 사이버 훈련장[1], 국방과학연구소의 사이버 훈련장[17] 등 민관군 분야의 사이버 훈련장 구축 및 운영이 활발히 진행되고 있다.

사이버 방어 훈련을 운영하는 주체는 크게 공격자를 의미하는 RT(Red Team)와 방어자를 의미하는 BT(Blue Team)로 나눌 수 있다[2]. 그 외, 역할별로 세분화하여 WT(White team), GT(Green Team)로 구성된다. BT는 사이버 방어 훈련에서 기관 내 시스템과 네트워크 등의 보호 대상 자산에 대한 방어활동을 수행하는 훈련의 주체를 의미한다. RT는 사이버 방어 훈련에서 BT의 방어 역량을 강화하기 위한 목적으로, 네트워크, 시스템을 대상으로 하여 다양한 공격자 행위를 모사하는 임무를 수행한다. 일반적으로 RT는 취약점을 공격하기 위한 침투테스트와 달리 공격자의 기술, 전술, 절차 및 목표를 에뮬레이션하여 네트워크 전체 방어 상태를 평가하는 임무를 수행한다[5]. WT는 전체적인 훈련 시나리오 설계 등의 훈련 관리자 역할을 담당하고, GT는 BT와 RT가 훈련을 수행하기 위한 기반 네트워크와 시스템 구축 업무를 수행하는 역할이다.

다수의 블루팀이 참가하고 방어 대상 시스템이 많은 실제 기관망을 모사한 대규모 훈련을 운영하기 위해서는 하나의 RT 프레임워크만을 사용해서는 쉽게 블루팀에 의해 탐지될 수 있으므로 도전적인 훈련을 구성하기 어렵다. 훈련 참가자들이 몰입할 수 있는 도전적인 훈련장 구성을 위해서는 다양한 공격 패턴, 네트워크 페이로드, 시스템 이벤트를 생성할 수 있도록 공격을 모사할 수 있어야 한다. 그러나 다양한 시나리오 기반의 공격 모사를 위해 수십 개 이상의 RT 프레임워크를 운영할 경우에 RT는 각 RT 프레임워크 마다 설정 및 모니터링해야 하며, C2(Command and Control) 서버들은 개별적으로 스코어 서버와 연동기능을 포함해야 한다. 이러한 C2 서버가 수십 개 이상이면 일일이 설정하기가 번거롭고, 사람이 운영하는 일이므로 실수가 발생할 여지가 있다.

본 논문에서는 다수의 공개용 RT 프레임워크와 직접 제작한 RT 프레임워크 등을 통합하여 대규모 훈련을 자동으로 운영할 수 있는 통합 RT 프레임워크(iRF: Integrated Red team Framework)를 제안한다.

II. 관련 연구

2.1 RT 프레임워크

레드팀 구성(Red Teaming)을 통한 테스트는 전문적인 지식을 가진 인원에 의해 수행되므로 비용이 많이 들고 반복적으로 수행해야 하는 문제가 있어 일관성 있게 수행하기 어렵다[5]. 이러한 이유로 RT 프레임워크를 개발하여 이를 자동화하려는 연구들이 진행되었다[3,4,5]. RT 프레임워크는 RT의 행위를 자동화하기 위한 도구이며, 특히, 초기 접근 단계 이후에 공격자의 명령을 받아서 처리하는 Post-Exploitation 행위를 주로 관장하므로 C2 프레임워크라고도 불린다. RT 프레임워크는 사이버 방어 훈련을 위한 목적으로도 사용되지만, 해커들도 공격에 광범위하게 활용한다. 이렇듯이 공개 RT 프레임워크들을 활용하여 사이버 훈련을 구성하는 것이 실제 공격에 대한 대응 연습이 될 수 있다고 할 수 있다.

Fig.1은 2019년 10월 1일부터 2020년 9월 30일 사이에 발생한 침입 사례에서 발견된 악성코드 패밀리의 활용 빈도를 보여준다[7]. 악성코드 패밀리는 악성코드의 기능 및 특징으로 그룹화한 것이다. 해커그룹에서 사용하는 악성코드도 수정 및 기능 보완을 통해 재활용된다. 사이버 킬체인 단계에서 다양한 접근 경로를 통해 침입한 후에 Post-Exploitation 행위를 수행하는 에이전트 프로그램을 통하여 침입한 영역 내에 가치 있는 정보를 수집하고 다른 시스템 및 네트워크로 이동하거나 정보유출, 랜섬웨어, 서비스 중단 등의 목표 행동을 수행한다. Fig.1의 BEACON은 상용 RT 프레임워크인 Cobalt Strike의 에이전트 프로그램을 일컫는다. Empire는 소스가 공개된 파워쉘 기반 에이전트와 파이썬 기반의 서버로 구성된 공개용 RT 프레임워크이다.

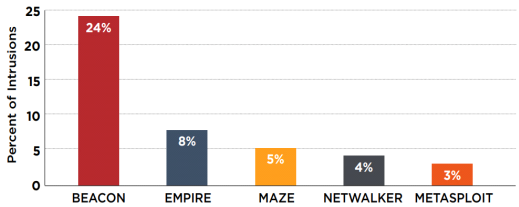

Fig. 1. Most Frequently Seen Malware Families, 2020 [7]

Table 1. Comparison of RT Framework[4]

C2	Server Language	Agent Language	Multi-User	UI	API
Apfell	Python	Python	○	Web	○
Caldera	Python	Go	○	Web	○
Cobalt Strike	Java	Java	○	GUI	✘
Covenant	C#	C#	○	Web	○
Dali	Python	Python	✘	CLI	✘
Empire	Python2	PowerShell	✘	GUI	○
EvilOSX	Python	Python	✘	GUI	✘
Faction C2	.NET	.NET	○	Web	○
FlyingAFalseFlag	Python	C++	✘	CLI	✘
godoh	Go	Go	✘	CLI	✘
ibombshell	Python	Powershell	✘	CLI	✘
INNUENDO	Python	Python	○	Web	○
Koadic C3	Python	JScript/VBScript	✘	GUI	✘
MacShellSwift	Python	Swift	✘	CLI	✘
Metasploit	Ruby	C/Java/PHP/Python	○	CLI	○
Merlin	Go	Go	✘	GUI	✘
Nuages	Python	C#	○	GUI	○
Octopus	Python	Powershell	✘	GUI	✘
PoshC2	Python	PowerShell/C#/Python	○	CLI	○
Prismatica	Javascript/Python	JScript/.NET/Rust	○	GUI	○
PowerHub	Python	PowerShell	○	Web	✘
Red Team Toolkit	Python	C++	✘	CLI	✘
ReverseTCPShell	PowerShell	PowerShell	✘	CLI	✘
SCYTHE	Python	C	○	Web	○
SilentTrinity	Python	IronPython	○	CLI	✘
Sliver	Go	Go	○	CLI	✘
Trevor C2	Python	Python/PowerShell	✘	CLI	✘
Weasel	Python	Python	✘	CLI	✘

RT 프레임워크에는 Metasploit, Cobalt Strike, Empire, Caldera, Apfell 등의 상용 또는 무료 도구들이 있다[4].

Zilberman[3]은 오픈소스 위협 에뮬레이션 도구 11개를 기능과 운영 측면으로 크게 나누고 기능 측면에서는 시나리오 정의 항목, 실행 항목으로, 운영 측면에서는 운영 시 전문지식 요구 정도 등의 운영자 관련 항목과 OS 호환성 등의 환경 관련 세부적인 비교 결과를 제공하여 RT 프레임워크를 사용할 때 목적과 필요에 따라 선택할 수 있도록 제공한다.

Table. 1은 서버 언어, 에이전트 언어, 멀티유저 지원 여부, 제공되는 사용자 인터페이스 및 API 제공 여부 등의 특징을 기반으로 공개 및 상용 RT 프레임워크들을 비교한다[4]. 특히 API 제공 여부는 각 RT 프레임워크의 기능을 외부에서 제어할 수 있도록 REST API와 같은 호출 인터페이스를 제공하는 것으로써 통합 RT 프레임워크 개발 시에 유용한 기능이다.

2.2 통합 RT 프레임워크의 필요성

일반적으로 사이버 방어 훈련에서 RT 프레임워크 운영 매커니즘은 Fig.2와 같다. BT 머신에 설치된 에이전트 프로그램은 주기적으로 C2 서버쪽으로 Beacon 메시지를 보내고 명령을 수신하기 위해 대기한다. RT는 CLI(Command Line Interface) 또는 GUI(Graphical User Interface)를 통해 에이전트에서 수행할 명령들을 설정한다. 일반적으로 EDR, SIEM 등의 보안 솔루션 제품 테스트에는

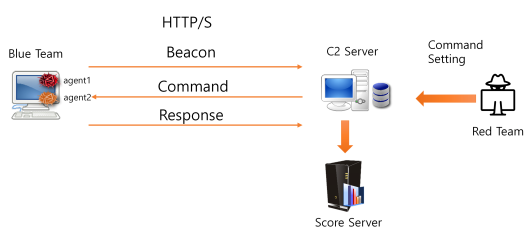

Fig. 2. Single RT Framework (Model 1)

Model 1 형태로도 충분히 가능하다. 그러나, 훈련 참가자들의 자유도가 높은 사이버 방어 훈련 환경에서는 단일 형태의 RT 프레임워크를 사용할 경우 C2 서버 IP 차단만으로 훈련 진행이 멈출 수 있으므로, Fig.3 Model 2 이상을 사용해야 한다. Fig. 3은 다수의 공개용 RT 프레임워크를 활용하여 훈련 시나리오를 구성한 모델이다. 공격자 페이로드 다양화, 에이전트와 C2 간 명령 채널 및 프로토콜 다양화를 통해 IoC(Indicator of Compromise, 침해지표)를 다양하게 구성할 수 있다는 장점이 있으므로 방어자에게 경험을 풍부하게 제공하기 위해서라도 한 가지만 사용하기보다는 되도록 여러 가지를 혼용하여 구성하는 것이 좋다.

수십 개 이상 다수의 RT 프레임워크를 이용하여 대규모 사이버 방어 훈련을 설계 및 운영하는 경우, RT는 각 RT 프레임워크의 GUI, 또는 CLI에 명령 설정 등의 입력과정을 직접 수행해야 한다. 각 RT 프레임워크에 속한 에이전트들의 명령 수행 결과에 따라 점수 서버와의 연동 작업이 필요하다. 이러한 이유로 각 RT 프레임워크들을 통합하여 관리할 수

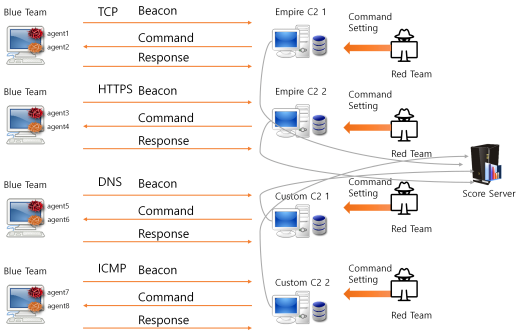

Fig. 3. Multiple RT Frameworks (Model 2)

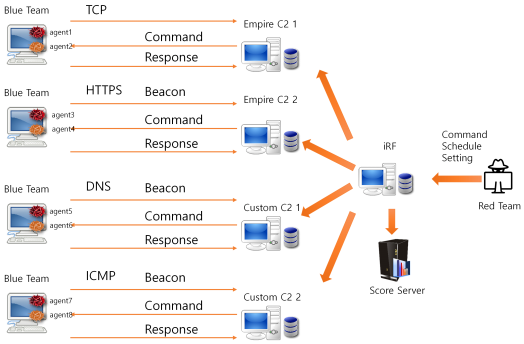

Fig. 4. Integrated RT Frameworks (Model 3)

있는 Fig.4와 같은 통합 RT 프레임워크를 개발하여 번거로운 작업을 방지할 수 있다.

RT 프레임워크를 통합한 사례로 Cobalt Strike에 Empire를 통합한 Beaconpire[9]가 있다. Cobalt Strike의 확장 스크립트(Aggressor Script)[10]를 추가하여 Empire와 Cobalt Strike 에이전트 간의 세션 패싱 기능과 Empire 에이전트 관리 기능이 있는 것으로 보이나 Cobalt Strike의 GUI에 Empire 에이전트 관리기능을 추가한 것으로 명령 일정 관리 기능이 없고, 훈련시마다 프레임워크에 속한 에이전트 목록에서 RT의 명령 입력 등의 수작업은 여전히 필요한 것으로 보이며, 다른 RT 프레임워크의 통합은 다루지 않고 있다.

III. 대규모 훈련을 위한 통합 RT 프레임워크 기능

2장에서 언급하였듯이, 다수의 RT 프레임워크를 혼용하여 사용하는 환경에서는 각 RT 프레임워크를 구성하는 C2 서버들을 일일이 설정하기가 번거롭고, 사람이 운영하는 일이므로 실수가 발생할 여지가 있다. 또한 훈련 평가를 위해서는 모든 C2 서버에 점수화를 위한 기능을 추가해야 하는 번거로움이 있다. 따라서 본 논문에서는 대규모 훈련을 위한 통합 RT 프레임워크에 아래와 같은 기능들을 제공한다.

- 대규모 에이전트 일괄 명령 전달 기능
- 명령 스케줄 기능
- 증거 기록 기능
- 점수화 기능

이러한 기능들에 대한 세부사항은 다음 각 절에서 소개된다.

3.1 대규모 에이전트 일괄 명령 전달 기능

실제 공격자들은 RDP, 리버스 쉘, RAT (Remote Administration Tool) 등을 통해 CLI 또는 GUI 형태로 피해 시스템에 접속, 관찰하면서 내부 정보 수집 및 컨트롤을 수행할 것이다. 훈련 환경에서의 RT는 공격자의 행위를 모사하지만 시스템마다 접속하여 콘솔에서 명령을 입력하는 방식으로 작업할 인력과 시간의 한계가 있다. 따라서 접속하는 에이전트들의 현황을 파악하고, 임의의 작업을 수행하도록 일괄 명령을 전달할 수 있는 기능이 필수적이다. 아무리 기능이 탁월하고 다양한 익스플로잇을 보

유하고 있는 도구일지라도 일일이 에이전트별로 콘솔에 입력하면서 정찰 등의 명령 전달행위를 해야 한다면 활용을 보류해야 할 것이다.

3.2 명령 스케줄 기능

명령 스케줄 기능은 훈련 시나리오별 공격 체인을 구성하기 위해 필수적이다. APT 공격자의 공격 단계는 록히드마틴사의 사이버 킬체인이라는 개념으로 널리 알려진 7가지 단계로 구성된다[11]. 시나리오 구성 시에는 단계별로 공격자의 전술, 기법, 절차를 실제 APT 해커그룹의 사례 분석 정보를 기반으로 더욱 세부적으로 정의하고 기법별 탐지 및 완화 방법까지 지식베이스로 구축한 MITRE의 ATT&CK[8] 매트릭스를 기반으로 공격자의 TTP(Tactics, Techniques, Procedures)를 정의하고 구성하는 기능이 필요하다. 공개된 RT 프레임워크들의 경우 CLI 형태로 실시간 명령 실행 기능은 있지만, 공격자 행위를 모사하여 훈련 시나리오 구성을 위해 필수적인 스케줄에 따른 절차적 명령 수행 기능이 없는 경우가 많다. 대규모 사이버 방어 훈련을 자동화하기 위해서는 명령 스케줄 기능이 필요하다. Fig.5는 공급망 공격을 통한 공격 시나리오를 ATT&CK 매트릭스 기반으로 구성한 예이다. 이러한 공격 시나리오를 구성하기 위해서는 에이전트가 수행할 명령들의 순서와 시간 설정 기능이 필요하다.

3.3 증거 기록 기능

명령 실행의 결과로 즉, 명령 실행 전 탐지 및 차단 실패의 결과로 훈련 점수에 반영됨과 함께 그에 대한 증거자료도 보존되어야 한다. BT 시스템에서의 악성 행위에 의한 결과 산출물은 날짜, 시간, IP, MAC 주소 정보, 화면캡처 등과 함께 C2 서버에 파일 또는 DB에 기록을 저장하여 방어 실패에 대한 증거자료로 작용할 수 있다. 추후 BT에서 이의 신청을 하는 경우 언제든지 비교 검토가 가능하도록 제시할 수 있어야 한다.

3.4 점수화 기능

이상징후에 대한 미탐지로 인하여 악성 행위가 지속되는 팀의 경우, 점수 서버에서 감점이 발생한다. BT는 실시간으로 점수 상황을 모니터링하면서 특정 시스템 및 네트워크 단의 방어 전략을 수립 및 보완한다. 따라서 공격자 명령으로 인한 피해 발생 시(RT 입장에서는 공격이 성공하는 경우) 점수 서버에 즉각적으로 반영되어야 한다. 그러나, APT 공격에 대한 방어 훈련의 경우에는 사이버 킬체인의 초기 단계에서 점수를 감점하는 경우, 보호대상 시스템과 네트워크 상의 이벤트 분석을 통한 탐지가 아닌, 점수 감점에 의한 탐지가 될 수 있으므로 방어 활동으로 탐지할 수 있도록 어느 정도 시간 경과 후에 감점할 수 있는 기능이 필요하다.

또한, BT의 시스템 리부팅, 일시적인 네트워크 설정 오류 상황 등으로 인하여 점수 감점이 발생하지

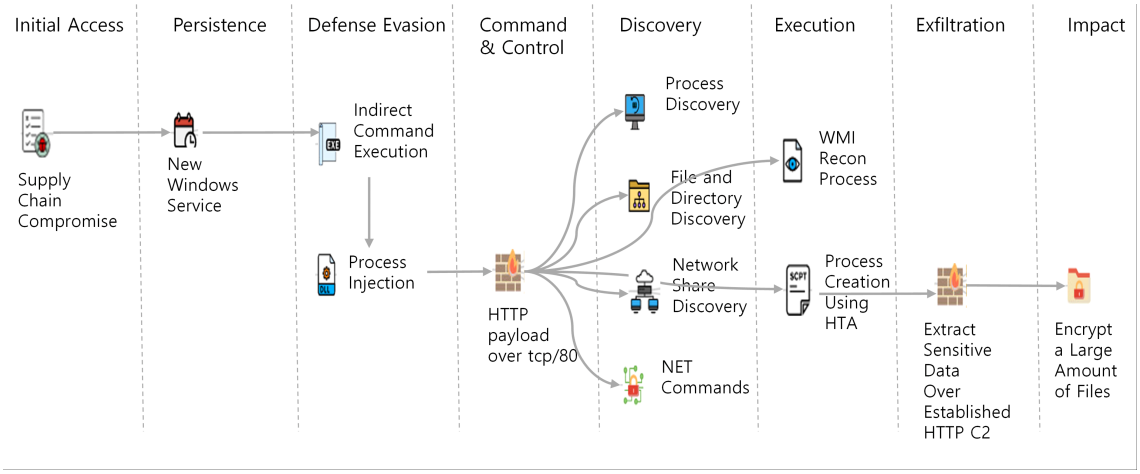

Fig. 5. An Example of Intrusion Chain

않을 경우, 공정성 문제가 발생할 수 있으므로 이를 종합적으로 고려하여 점수화할 수 있는 시스템이 필요하다.

IV. iRF 구현

본 논문에서는 Empire, Metasploit 등의 공개 또는 상용으로 활용 가능한 RT 프레임워크 및 직접 제작한 RT 프레임워크 등 다수의 RT 프레임워크들을 통합하여 운영할 수 있는 통합 RT 프레임워크인 iRF (Integrated Red team Framework)를 설계 및 구현하였다. 구현 환경은 다음과 같다.

- OS : Ubuntu Linux 20.04
- Web Server : Apache Tomcat
- Server Language : Java, JSP
- DB : Mysql

웹서버와 DB를 Docker로 구성하여 Ubuntu 가상머신에 구축하고, 가상화 서버는 VMWare vSphere ESXi 6.7에서 개발하였다. iRF는 일련의 C2 서버들(C2 Pool)과 상호작용하고 훈련 결과를 점수 서버에 전달한다. RT는 공격 시나리오를 구성하기 위해 명령 설정 및 명령 실행 스케줄을 입력하도록 구현하였다. 명령 설정을 위한 사용자 인터페이스는 JSP로 구현된 웹 GUI 형태로 멀티유저를 지원한다.

RT는 iRF에만 명령 스케줄을 설정하고(Fig.6의 ①) 다수의 RF 프레임워크마다 설정할 필요가 없다. 즉, RT 프레임워크를 구성하는 C2 서버가 Empire이든, Caldera이든 상관없이, 주어진 훈련 스케줄에 따라 일관된 명령을 전송할 수 있다.

iRF는 설정된 스케줄에 따라 등록된 C2 서버들에 명령을 전파하고(Fig.6의 ③) 결과를 받아서(Fig.6의 ④) DB에 저장한다(Fig.6의 ⑥). 그리고, 결과를 취합하여 BT별로 점수를 계산하여(Fig.6의 ⑦) 스코어 서버에 전달한다(Fig.6의 ⑧).

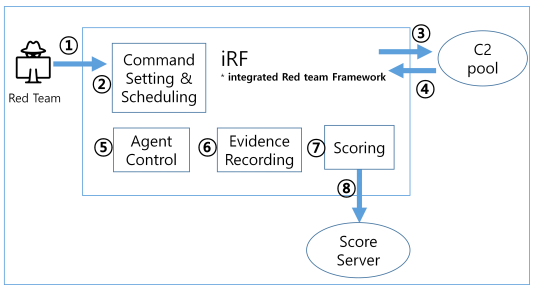

Fig. 6. Components of iRF

4.1 C2 등록

iRF를 구동하기 위해서는 관리할 RT 프레임워크의 C2 등록을 우선으로 수행해야 한다(Fig.6의 ①) C2 정보는 Empire, Caldera, Metasploit 등의 무료 또는 상용 C2와 훈련용으로 직접 제작한 C2들의 IP, Port 정보, 통신을 위한 프로토콜 정보 및 RT 프레임워크를 구분하기 위한 값인 Type, 계정 정보, Stale Time 설정 값 등을 등록하는 인터페이스를 가진다. 에이전트로부터 일정 기간 이상 연락이 없는 경우, 해당 에이전트는 명령을 수행하기 어려운 상태이므로 관리 대상에서 제외할 필요가 있다. Stale Time은 장기간 연락이 없는 에이전트들을 별도 표시하거나 제거할 때 기준이 되는 기간을 의미하며 구현 시 고려해야 한다.

4.2 에이전트 관리

에이전트 관리 기능은 3.1의 대규모 에이전트에 일괄 명령 전송 기능을 구현하기 위한 것이다. 개별 C2들에 연결된 에이전트들의 정보를 주기적으로 조회하여 DB에 저장하고 iRF의 웹기반의 대쉬보드에 표시한다. RT는 대쉬보드에서 에이전트들의 현황을 한눈에 확인할 수 있다. 에이전트들은 iRF와 통신할 필요 없이, 각 에이전트가 소속된 C2 서버들과 명령 전달 체계를 유지한다. iRF는 C2들의 정보들만 주기적으로 조회 및 설정함으로써(Fig. 6의 ③) 각 RT 프레임워크에 속한 에이전트들과 통신할 필요 없이 에이전트 상태 확인 등이 가능하다.

4.3 공격 체인 구성을 위한 명령 스케줄

MITRE ATT&CK의 공격 기법들을 구성하기 위한 명령은 배치파일 또는 파워쉘 등과 같은 스크립트 언어의 한 줄 분량으로 구성할 수 있는 단순한 유형과 랜섬웨어나 파일 유출 등의 전술을 위한 기법들과 같이 한 줄 분량으로는 구성할 수 없는 스크립트 파일 또는 실행파일 형태의 별도 프로그램으로 제작되어야 하는 복잡한 유형으로 나눌 수 있다. 이러한 단순 유형 또는 복잡 유형의 개별 명령들은 공격 시나리오를 구성하기 위해 시작 시각, 반복 횟수, 종료

시각 등을 설정하여 공격 체인 구성을 위한 재료로 활용된다. Table.2는 Fig.5의 공격 체인 구성 예에 포함된 공격자 기법에 대한 상세 정보를 표시한다. 동일한 기법이라도 다수의 TEST CASE를 포함할 수 있는데, 구체적으로 선정된 TEST CASE를 지정함으로써 좀 더 다양한 훈련을 수행할 수 있다. 예를 들면, Discovery 전략을 위한 공격 기법 중 Process Discovery 기법의 경우 "tasklist" 명령을 통해 수행할 수도 있지만, 파워쉘의 "Get-Process" 명령, 또는 "WMIC" 명령을 통해 수행할 수도 있다.

현실 세계에서의 공격자는 목표한 행동을 수행하기 위해 최대한 탐지되지 않도록 하는 것이 중요하다. 그러나 RT의 경우에는 BT의 탐지 및 방어 활동을 돕기 위해 일정 기간 반복적으로 수행하는 것이 필요하다. 초기에 탐지에 실패하더라도 1차 공격에서 발생한 공격 이벤트 로그 분석, 네트워크 트래픽 분석 및 방화벽, IPS(Intrusion Prevention System) 로그 분석 등을 통해 탐지 및 방어를 수행하면 2차 공격 시에는 탐지 및 차단에 성공할 수 있으며 감점을 당하지 않을 수 있다. 이를 위해 자동화된 공격 체인 설정 및 반복 설정 기능을 구현하였다.

iRF에서 C2별로 Agent 목록을 얻어서 등록된 명령 스케쥴에 따라 명령을 전달하고 결과를 저장하는 전체 알고리즘은 Fig.7에서 설명한다.

```
c2s ← get_c2_list();
foreach c2info ∈ c2s do
  if is_api_supplied(c2info) then
    while (true) do
      token ← get_token(c2info);
      agents ← get_agents(c2info, token);
      foreach agent ∈ agents do
        if is_new(agent) then
          register_agent(agent);
          assign_schedule(agent);
        end
        set_agent_lastseentime(agent, current_time);
        commands ← get_scheduled_commands(agent);
        foreach command ∈ commands do
          execute_command(c2info, agent, command,
                          token);
          set_command_fetched_state(agent, command)
        end
        sleep(time)
        commands ←
              get_fetched_state_commands(agent);
        foreach command ∈ commands do
          result ← get_command_result(c2info, agent,
                                      command, token);
          set_result(agent, command, result);
        end
      end
    end
  end
end
```

Fig. 7. Algorithms of Agents Command Control

Table. 2. MITRE ATT&CK Techniques and Test Cases Included in Fig.5.

Phase	Technique	Test Case
Initial Access	T1195 - Supply Chain Compromise	Supply Chain Compromise(Korean SW #1)
Defense Evasion	T1218 - Signed Binary Proxy Execution	Process Creation Using HTA
Persistence	T1543 - Create or Modify System Process	Windows Service
Defense Evasion	T1055 - Process Injection	Process Injection via PowerSploit
Discovery	T1057 - Process Discovery	Process Discovery - tasklist
Discovery	T1083 - File and Directory Discovery	File and Directory Discovery
Execution	T1047 - Windows Management Instrumentation	WMI Reconnaissance Processes
Discovery	T1135 - Network Share Discovery	Network Share Discovery command prompt
Exfiltration	T1041 - Exfiltration Over Command and Control Channel	Extract sensitive data over established HTTP C2
Defense Evasion	T1202 - Indirect Command Execution	Indirect Command Execution
Command & Control	T1071 - Application Layer Protocol	Empire Default Beacon HTTP payload over tcp/80
Discovery	T1087 - Account Discovery	NET Commmands
Impact	T1486 - Data Encrypted for Impact	Encrypt a Large Amount of Files

4.4 명령 수행 결과 저장 및 표출

iRF의 명령 스케줄에 따라 각 C2 서버에 전달된 명령은 에이전트로 전달되고 에이전트가 수행한 명령 결과는 비동기적으로 각 C2 서버에게 전달된다. C2 서버들은 iRF에서 전달받은 명령을 각 에이전트들에게 전달하고 그 수행 결과를 파일 또는 DB에 저장한다. iRF는 주기적으로 명령 결과들을 C2 서버들에게 요청하고, 수신한 결과들을 DB에 저장한다. BT의 방어활동 진행 상황 파악을 위해 에이전트가 명령을 fetch하였고 그 결과를 받은 경우(결과 수신 성공), fetch는 하였으나 결과를 받기 전 차단된 경우(결과 수신 실패), 또는 fetch 하기 전 차단을 통해 fetch 하지 않은 경우(fetch 실패)를 상호 구분하여 표시하도록 구현하였다.

Fig.8과 Fig.9는 각각 공개 C2 서버에 저장된 에이전트의 명령 수행 결과 파일과 iRF에 저장된 에이전트 명령 수행 결과 표출 화면이다. RT는 Fig.8과 같은 화면을 보기 위해 각 C2 서버 시스템의 콘솔 또는 GUI에 접속하여 명령 수행 결과를 확인할 필요 없이 Fig.9와 같이 iRF 대쉬보드에서 BT 시스템별 Agent별 명령 수행 상황을 모니터링하고, BT의 방어 상태를 파악하여 공격 반복 주기, 공격 효과를 위한 행위 등을 동적으로 설정할 수 있다. 즉, BT가 전혀 탐지를 못 할 때는 공격 효과를 보여주기 위한 행위를 보다 앞당겨서 반복적으로 실행하여 분석의 실마리를 제공해 줄 수 있다. 침해사고의 근본적인 원인이 되는 취약점 또는 악성코드의 근본적인 감염 원인 파악을 통한 방어가 아닌 수상한 C2 서버의 IP를 방화벽에서 차단하여 방어를 수행하는 경우에는 추가적인 악성코드 다운로드를 통한

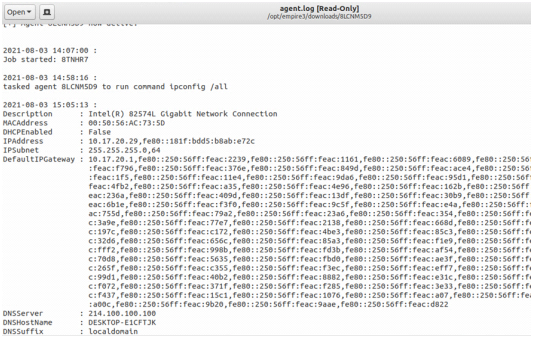

Fig. 8. An Example of Command execution result log file stored in a Open C2 server

Fig. 9. An Example of Command execution result stored in iRF

재감염 등의 명령을 전달하는 조치를 할 수 있다.

4.5 점수 서버 연동

점수는 수행한 명령의 심각도에 따라 다르게 설정할 수 있어야 한다. 시스템의 유용한 정보를 발견하기 위한 정찰 명령들과 비교해 파일 유출, 서비스 중단, 파괴 등의 명령 행위에 점수 가중치를 줄 수 있다. 주요 침해행위가 발생하기 전에 정찰 단계에서 탐지 및 차단에 성공한다면 더 큰 감점을 당하기 전에 차단함으로써 점수를 높게 받을 수 있도록 구현하였다. 다수의 RT 프레임워크를 사용할 경우에는 C2 서버마다 점수 서버와 연동하도록 구현해야 하는 단점이 있다, 그러나, iRF는 각 C2들의 정보를 취합하여 행위별 점수 가중치, 점수 반영 시간대 등을 일괄 설정 및 반영할 수 있도록 구현하였으며, 각 C2 서버들이 점수 서버와 연동하도록 구현할 필요가 없다는 장점이 있다.

4.6 실험 결과

본 연구에서 개발한 iRF는 2021년 8월 20일 실제 훈련에 사용되었다. 총 4개의 APT 훈련 시나리오를 설정하였으며, 50여 개 팀이 3시간 동안 사이버 방어 훈련에 동시 참여하였다. 이를 위해 시나리오가 주입된 200개의 BT용 가상머신이 생성되었으며, 각 가상머신에는 2개 이상의 RT 프레임워크에 소속된 에이전트 프로그램들이 명령 일정에 따라 동작하도록 구성하였다. 이상징후에 대한 탐지 및 차단이 전혀 수행되지 않을 경우, -1000점을 받도록 구

성하였으며, 정찰 행동은 -10점, 파일과 계정 정보 유출, 랜섬웨어 등의 행동은 -40점으로 구성하여 훈련을 운영한 결과, BT별로 정확히 점수가 반영됨을 확인하였다.

V. 결 론

최근 제어시스템 공격, 공급망 공격, 랜섬웨어 등 사이버 공격이 더욱 지능화되고 있어 각 기관의 사이버보안 담당자들의 탐지 및 대응 역량 제고가 중요한 시점이다. 이를 위해서는 평시에 실제 공격과 유사한 다양한 공격 기법에 대한 탐지 및 차단, 보안강화 훈련을 반복적으로 수행하는 것이 무엇보다 중요하다 할 수 있다. 이러한 추세에 맞추어 전문 인력의 수작업을 최소화한 자동화된 RT 프레임워크 개발 연구들이 많이 진행되고 있다. 기존 연구들에서 공개용 RT 프레임워크 비교 평가를 수행하는 등 수많은 RT 프레임워크 중 각자에게 필요한 기능을 선정하는데 도움을 주고 있다[3,4].

사이버 방어 훈련을 보다 도전적으로 운영하기 위해서는 특정 RT 프레임워크를 선정하여 훈련장을 구성하기보다는 공개 및 자체 제작 RT 프레임워크를 혼용하여 수십 개의 C2 서버들을 운영하는 훈련장을 구축하는 것이 BT의 방어 경험을 다양화할 수 있다는 측면에서 유리하다. 그러나 수십 개의 C2 서버를 운영하는 데는 전문지식을 가진 인력의 반복적인 수작업을 동반한다는 문제점이 있다. 본 연구에서는 이러한 문제를 해결하기 위해 통합 RT 프레임워크인 iRF를 개발하여 RT의 수작업을 최소화하였다.

현재는 REST API를 지원하는 일부 공개 C2와 자체 제작 C2들만을 지원하는 형태이지만, 향후에는 API를 지원하지 않는 C2의 데이터베이스 또는 로그 파일을 분석하고 이를 API로 추가하는 연구를 수행할 예정이다.

References

[1] Younghan Choi, et al. "Design and Implementation of Cyber Range for Cyber Defense Exercise Based on Cyber Crisis Alert." Journal of the Korea Institute of Information Security & Cryptology 30(5), pp. 805-821, Oct. 2020.

[2] Brilingaitė, Agnė, Linas Bukauskas, and Aušrius Juozapavičius, "A framework for competence development and assessment in hybrid cybersecurity exercises," Computers & Security, vol. 88, pp. 1-13, Jan. 2020.

[3] Zilberman, Polina, et al. "SoK: A Survey of Open-Source Threat Emulators," arXiv preprint arXiv:2003.01518, 2020.

[4] Matrix, "C2 Matrix," https://www.thec2matrix.com/matrix, last accessed Aug. 2021.

[5] Applebaum, Andy, et al. "Intelligent, automated red team emulation," Proceedings of the 32nd Annual Conference on Computer Security Applications, pp. 363-373, Dec. 2016.

[6] Empire, "Powershell Empire", https://www.powershellempire.com, last accessed Aug. 2021.

[7] FireEye, "M-Trends 2021," https://content.fireeye.com/m-trends-kr/rpt-m-trends-2021-kr, last accessed Aug. 2021.

[8] MITRE ATT&CK, "ATT&CK," https://attack.mitre.org/matrices/enterprise, last accessed Aug. 2021.

[9] Bluescreenofjeff.com, "Beaconpire," https://bluescreenofjeff.com /2016-11-29-beaconpire-cobalt-strike-and-empire-interoperability-with-aggressor-script/, last accessed Aug. 2021.

[10] CobaltStrike, "Aggressor Script," https://www.cobaltstrike.com/aggressor-script/, last accessed Aug. 2021.

[11] Hutchins, Eric M., Michael J. Cloppert, and Rohan M. Amin, "Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains," Leading Issues in Information Warfare & Security Research, vol. 1, pp. 80-106, Mar. 2011.

[12] The NATO Cooperative Cyber Defence Centre of Excellence, "Locked Shields," https://ccdcoe.org/exercises/locked-shields/, last accessed Aug. 2021.

[13] European Union Agency For Cybersec

urity, “Cyber Europe,” https://www.enisa.europa.eu/topics/cyber-exercises/cyber-europe-programme/cyber-europe-2022/, last accessed Sep. 2021.
〔14〕 Cybergym, https://www.cybergym.com, last accessed Sep. 2021.
〔15〕 Swedish Defence Research Agency, “CRATE-cyber range and training environment,” https://www.foi.se/en/foi/resources/crate---cyber-range-and-training-environment.html, last accessed Sep. 2021.
〔16〕 KISA Academy, “Security-Gym,” https://academy.kisa.or.kr/edu/apply10.kisa, last accessed Aug. 2021.
〔17〕 Myung Kil Ahn, Yong Hyun Kim, ″Research on System Architecture and Simulation Environment for Cyber Warrior Training,″ Journal of the Korea Institute of Information Security & Cryptology 26(2), pp. 533-540, Apr. 2016.

〈저자소개〉

장 인 숙 (In Sook Jang) 정회원
1998년 2월: 경북대학교 문헌정보학과 학사
2001년 2월: 경북대학교 컴퓨터과학과 석사
2015년 2월~현재: 충남대학교 컴퓨터공학과 박사과정
2001년 3월~현재: 한국전자통신연구원 부설연구소 책임연구원
〈관심분야〉 사이버보안, 사이버훈련, 정보보안 교육

조 은 선 (Eun-Sun Cho) 정회원
1991년: 서울대학교 계산통계학과 학사
1993년: 서울대학교 전산과학과 석사
1998년: 서울대학교 전산과학과 박사
2000년~2010년: 한국과학기술원 연구원
2002년~2006년: 충북대학교 조교수
2006년~현재: 충남대학교 컴퓨터공학과 교수
〈관심분야〉 프로그래밍언어, 프로그램 런타임 환경, 바이너리코드 분석