원격 코드 실행 공격 분석을 위한 Base64 디코딩 시스템

정장현[1], 정지환[2], 최성곤[2*]

[1]제이제이솔루션, [2]충북대학교

jjsol210120@gmail.com, dean987@naver.com, *choisg@cbnu.ac.kr

Base64 Decoding System for RCE Analysis

Jang Hyeon Jeong[1], Jeong Ji Hwan[2], Seong Gon Choi[2*]

[1]JJSolution Inc. , [2]Chungbuk National Univ.

요 약

본 논문은 원격 코드 실행 공격 분석을 위한 Base64 디코딩 시스템을 제안한다. 보안 인력들이 실제 보안 이슈에 대응하는 인력은 절반도 되지 않는다. 현재 보안 관제의 업무 비중은 오탐처리에 소요되는 작업량에 가장 많은 시간을 사용한다. 오탐 처리나 분석을 위한 정보수집이 오래 걸리는 이유는 다양하지만 그중 하나의 이유는 공격이 암호화되어 수신된다는 것이다. 부족한 보안 인력들의 실제 보안 이벤트에 대응 시간을 확보하기 위해 분석 작업량을 줄일 수 있는 원격 코드 실행 공격 분석을 위한 Base64 디코딩 시스템의 구성 및 실제 공격(Log4jshell) 패킷을 활용한 실험 결과를 제시한다.

Ⅰ. 서 론

최근 네트워킹 기술이 발전하고 보편화로 인하여 네트워크에는 보안에 민감한 데이터들이 존재하게 되었다. 이에 따라 데이터 보호를 위해 네트워크 보안이 중요해지고 있으며, 방화벽, IDS(Intrusion Detection System) and IPS(Intrusion Prevention System) 등 다양한 보안 시스템으로 네트워크 공격을 방어하려고 한다. [1],[2]

하지만 모든 보안 시스템들은 오탐이 존재하고 이를 구분하기 위한 인력, 시간, 노력이 필요하다. 하지만 2021년 기준 전 세계의 보안 인력은 350만 명이 필요하다고 한다. [3]

전 세계적으로 보안 인력이 매우 부족하지만 부족한 보안 인력들이 실제 보안 이슈에 대응하는 인력은 절반도 되지 않는다. 현재 보안 관제의 업무 비중은 오탐처리에 소요되는 작업량에 가장 많은 시간을 사용하는 것으로 확인되었다. 오탐처리에 소요되는 작업량이 45%, 분석을 위한 정보 수집 25%로 실제 보안 이벤트 분석 및 대응 업무는 30%밖에 되지 않았다. [4]

이렇게 오탐 처리나 분석을 위한 정보수집이 오래 걸리는 이유는 다양하지만 그중 하나의 이유는 공격이 암호화되어 수신된다는 것이다. 그림 1과 같이 공격이 평문으로 수신되면 보안 인력들은 해당 패킷의 페이로드만 확인하면 어떠한 공격인지 예상할 수 있다.

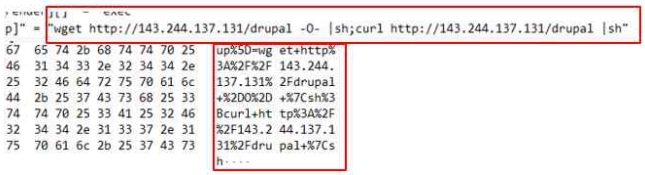

그림 1. 평문으로 된 원격 코드 실행 공격

하지만 그림 2와 같이 공격이 암호화되어 수신되면 해당 공격을 복호화하여 어떠한 공격이 시도되었는지 분석하고 대응해야 하므로 분석을 위한 노력이 더욱 필요하게 된다.

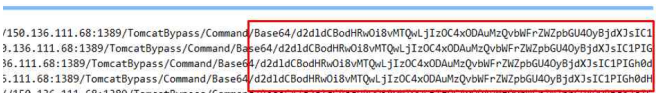

그림 2. 암호화된 원격 코드 실행 공격

따라서, 본 논문에서는 원격 코드 실행 공격 분석을 위한 Base64 디코딩 시스템을 제안한다. 제안하는 시스템은 보안 인력들의 분석 작업량을 줄여 실제 보안 이벤트에 대응 시간을 확보할 수 있도록 한다.

2장에서는 본 논문에서 제안하는 원격 코드 실행 공격 분석을 위한 Base64 디코딩 시스템의 구성을 설명한다. 3장에서는 제안된 시스템의 실험 결과를 제시하고 마지막으로 4장에서는 본 논문에 대한 결론을 도출한다.

Ⅱ. 제안하는 시스템

본 논문에서는 부족한 보안 인력들의 실제 보안 이벤트에 대응 시간을 확보하기 위해 분석 작업량을 줄일 수 있는 원격 코드 실행 공격 분석을 위한 Base64 디코딩 시스템을 제안한다.

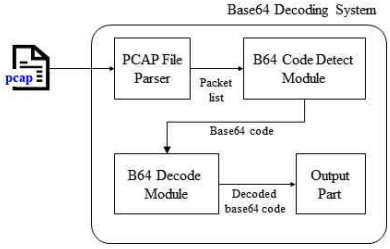

그림 3. 원격 코드 실행 공격 분석을 위한 Base64 디코딩 시스템

그림 3은 본 논문에서 제안하는 원격 코드 실행 공격 분석을 위한 Base64 디코딩 시스템의 구성도이다. 제안하는 시스템은 PCAP File Parser, Base64 Code Detect Module, Base64 Decode Module, Output Part로 구성되어 있다. PCAP File Parser는 입력(수신)된 PCAP 파일을 읽고 Base64 Code Detect Module로 패킷 단위로 전달한다. Base64 Code Detect Module은 수신된 패킷에 Base64로 인코딩된 문자열이 있는지 확

인하여 Base64로 인코딩된 문자열을 복호화하기 위하여 Base64 Decode Module로 전달한다. Base64 Decode Module은 수신한 base64 코드를 디코딩하여 Output Part로 전달하여 디코딩된 코드를 출력한다.

Ⅲ. Implementation

2021년도 12월에 전 세계적으로 큰 이슈가 되었된 로깅 도구 Apache Log4j의 취약점을 악용한 Log4shell(CVE-2021-44228)은 취약점 점수 최고치를 받은 공격이다. Log4shell 공격은 로깅 도구인 Log4j를 원격 공격 실행 코드를 로깅시켜 해당 코드를 실행시키는 RCE(Remote Code Execution) 공격의 일종으로 아주 위험한 공격이다. [5] - [7] 본 논문에서 제안하는 시스템을 테스트하기 위하여 Log4jshell 실제 공격 시도 패킷을 사용하였다. 그림 4는 실제 Log4jshell 공격 패킷의 페이로드 일부이다. 그림 4에서 해당 공격은 Base64로 암호화되어 실제 공격 코드를 확인할 수 없다. 실제 공격 코드를 확인하기 위해서는 추가적인 분석이 필요하다.

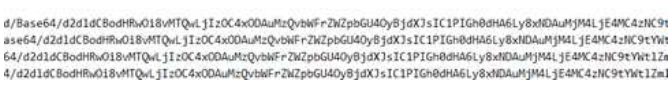

그림 4. 실제 Log4jshell 공격 패킷의 페이로드 일부

그림 4의 Log4jshell을 포함한 패킷들을 수신한 Base64 디코딩 시스템은 수신된 패킷 중에 base64로 암호화된 패킷이 존재하는지 탐색한다. Base64 Code Detect Module은 탐색 결과 base64 코드를 탐지하여 Base64 Decode Module에서 해당 코드를 복호화하도록 한다. Output Part에서는 그림 5와 같이 복호화된 RCE 코드를 출력하여 어떠한 공격이 시도되었는지 추가적인 분석 필요 없이 확인할 수 있도록 한다. 그림 5의 복호화된 명령어는 외부 IP(140.238.180.34)에서 특정 파일을 다운받고 최고 권한을 부여하여 해당 파일을 실행시키려는 시도로 확인되었다. 보안 인력은 외부 IP(140.238.180.34)에서 해당 파일을 다운로드한 시도가 있는지 확인을 하고 대응을 하면된다.

그림 5. 실제 Log4jshell의 RCE 코드와 복호화된 명령어

여기서 우리는 암호화된 공격을 복호화할 필요성을 확인할 수 있었다. 그림 6은 Output Part에서 추가로 출력해주는 정보로써 해당 패킷의 헤더의 내용이다.

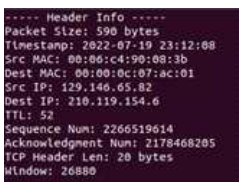

그림 6. 실제 Log4jshell 공격 패킷의 헤더 정보

해당 공격 패킷은 외부 IP(129.146.65.82)에서 수신된 공격이다. 하지만 그림 5와 같이 공격 패킷을 분석해보면 다른 외부 IP(140.238.180.34)에서 파일을 다운로드하는 명령어를 실행시킨다. 이 경우, 패킷이 수신된 외부 IP(129.146.65.82)보다 실제 공격이 이루어질 수 있는 공격 코드에 포함되어 있는 IP(140.238.180.34)와의 통신 이력을 확인할 필요성이 높다. 이처럼, 공격 패킷이 수신되는 IP와 실제 공격이 이루어지는 IP가 다를 수 있으므로 공격 코드에 대한 분석은 필수적이다.

Ⅳ. 결론

본 논문에서는 Base64 디코딩 시스템을 제안하였다. 제안한 시스템은 보안 인력들이 실제 공격에 대응할 수 있는 시간을 좀 더 확보하기 위해 공격 패킷 분석 시간을 단축하려는 목표가 있었다. 실제 공격 패킷으로 실험한 결과 base64로 암호화된 원격 코드 실행 코드를 자동으로 분석하여 출력함으로써 공격 분석 시간을 단축할 수 있었다.

네트워크 공격을 방어하기 위해 방화벽, 시그니처 기반의 IDS 및 IPS 등 다양한 보안 시스템을 사용하여 방어하려고 하지만 해당 시스템들은 기존에 알려진 공격만 탐지할 수 있다. 알려지지 않은 새로운 공격 즉, 제로데이 공격을 방어하기 위해서 AI를 통한 새로운 공격 탐지 및 분석 등 다양한 연구가 많이 진행되고 있다. [8]-[9] 하지만 AI를 공격 탐지도 오탐이 존재하며 추가적인 분석들도 필요하다. 제안한 시스템도 base64 암호화된 공격 코드를 자동으로 분석을 해주지만 실제 공격 성공 여부에 대한 공격은 보안 인력의 추가적인 분석이 필요하다. 따라서, 공격 분석을 자동화하기 위해 지속적으로 연구되어야 한다.

*교신저자: 최성곤(choisg@chungbuk.ac.kr)

ACKNOWLEDGMENT

본 연구는 과학기술정보통신부 및 정보통신기획평가원의 지역지능화 혁신인재양성(Grand ICT연구센터) 사업의 연구결과로 수행되었음" (IITP-2022-2020-0-01462). 또한, 이 논문은 2022년도 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (2019R1A2C1006167).

참 고 문 헌

[1] Bhavya Daya, "Network Security: History, Importance, and Future Gainesville", FL, USA, 2013.

[2] Jang Hyeon Jeong, Jong Beom Kim, Seong Gon Choi, "Zero-Day Attack Packet Highlighting System", International Conference on Advanced Communications Technology (ICACT) 2021, Feb. 2021.

[3] The Fullstack Academy Team, "Cybersecurity Job Demand: The Growing Need for Cyber Professionals", Jul. 2022

[4] IBM "AI, 사이버 보안을 위한 핵심 역량", Nov. 2018

[5] Shein Sopariwala, Enda Fallon, Mamoona Naveed Asghar, "Log4jPot: Effective Log4Shell Vulnerability Detection System", 2022 33rd Irish Signals and Systems Conference (ISSC), Jun. 2022

[6] Douglas Everson, Long Cheng, Zhenkai Zhang, "Log4shell: Redefining the Web Attack Surface" The Network and Distributed System Security (NDSS) Symposium 2022, Apr. 2022

[7] National Institute of Standards and Technology. NVD - CVE-2021-44228, 2021

[8] Nisreen Innab, Eman Alomairy, Lamya Alsheddi, “Hybrid System Between Anomaly Based Detection System and Honeypot to Detect Zero Day Attack”, 2018 21st Saudi Computer Society National Computer Conference (NCC), April 2018.

[9] Masaaki Sato, Hirofumi Yamaki, Hiroki Takakura, “Unknown Attacks Detection Using Feature Extraction from Anomaly-Based IDS Alerts”, 2012 IEEE/IPSJ 12th International Symposium on Applications and the Internet, July. 2012.