IT 서비스의 취약점 발생 원인과 대응 방안: 인공지능 기술의 양면성

장수혁*, 박재경O
*한국폴리텍대학교 사이버보안과,
O한국폴리텍대학교 사이버보안과
e-mail: papalooo@naver.com*, jakypark@kopo.ac.krO

A study on the causes and countermeasures of IT service vulnerabilities: Two sides of artificial intelligence technology

Su-Hyeok Jang*, Jae-Kyeong ParkO
*Dept. of Cybersecurity, Korea-Polytechnic University,
ODept. of Cybersecurity, Korea-Polytechnic University

● 요 약 ●

본 논문에서는 상용 소프트웨어나 웹, 앱, 클라우드 서비스 등 다양한 IT 서비스에서 취약점이 발생하는 근본적인 원인을 알아보고 그에 대한 효과적이고 미래지향적인 대응 방안을 제안한다. 이 대응 방안은 공개된 취약점들을 학습한 인공지능 모듈을 기존의 개발환경에 도입하는 것을 통해 개발 중인 서비스의 설계 문제에 대해 즉각적인 피드백을 줌으로서 작업 효율을 높이고 피드백한 취약점의 위험도를 함께 알려줌으로 혹여 미흡했을 수 있는 개발자의 기존 보안 의식 수준을 높여서 IT 시장에 전체적으로 긍정적인 영향을 끼칠 수 있을 것이라 보여진다. 이 과정을 통해 IT 보안 관점에서 인공지능의 양면성을 바라보고 점점 발전해 가는 인공지능 기술 앞에 우리가 각추어야할 자세를 제안하고자 한다.

키워드: 보안 의식 수준(security awareness level), 인공지능(Artificial intelligence)

I. Introduction

IT 서비스에서 취약점이 발견되는 것은 매우 큰 피해를 불러일으킬 수 있다. 실제로 상용화 된 후 취약점이 발견되어 해당 서비스를 사용중인 여러 플랫폼이 피해를 입는 경우도 여럿 발생했었다. 취약점이 발생하는 원인으로는 개발자가 의도치 않은 값이 입력될 때의 부적절한 외부 입력 처리나 운영 환경의 차이를 고려하지 않은 경우나 잘못된 메모리접근 등 여러 이유가 있지만 앞서 말한 취약점 발생 원인들은 결국 개발자가 부주의하게 서비스를 개발했기 때문에 발생한 것이다.

접근 등 여러 이유가 있지만 앞서 말한 취약점 발생 원인들은 결국 개발자가 부주의하게 서비스를 개발했기 때문에 발생한 것이다. 실력이 뛰어난 개발자라도 사람이기 때문에 항상 완벽할 순 없으므로 취약점이 발생하지 않는 것은 매우 힘들다.

본 논문에서는 최근 빠르게 발전하고 있는 인공지능 기술을 활용한 대응 방안을 제안할 것이며, 그 과정에서 IT 보안의 관점에서 인공지능의 양면성을 바라보고 우리가 지녀야 할 자세에 대해 생각해 볼 것이다.

II. Preliminaries

1. Related works

1.1 국내 동향

C++ 컴파일러는 컴파일 시간에 간접 호출 대상에 대한 제어 흐름을 분석한 다음 런타임에 대상을 확인하는 코드를 삽입하는 보안 기능과 악용될 위험이 있는 함수에 오버런 감지 코드를 삽입하는 보안 기능도 제공한다.

Java 컴파일러는 코드의 유효성을 검사하여 취약점을 식별하는 보안 기능을 제공한다. 예를 들어, Java 컴파일러는 null 포인터 참조와 같은 유효하지 않은 메모리 참조를 식별할 수 있다. 또한

Java 컴파일러는 코드를 최적화하여 메모리 오류가 발생할 가능성을 줄이는 보안 기능도 제공한다. rust 컴파일러는 소유권(ownership) 시스템을 사용하여 메모리 할당과 해제를 추적한다. 이를 통해 메모리 누수와 오버런과 같은 취약점을 줄인다. 그림 1과 같이 인공지능을 잘 활용하면 방어자 역할을 수행할 수 있으나 이를 악용할 경우 기존보다 더 강력한 공격자 역할을 할 수 있다는 것을 파악해야 할 것이다.

Table 1. Two sides of artificial intelligence technology

1.2 개선 방안

IT 서비스들의 기능을 이루는 소스 코드들은 다양한 언어로 작성돼있으며 각 언어마다 다른 컴파일러를 사용하고 있다. 한 컴파일러에 국한되지 않는 진단을 위해 인공지능 기술을 활용한 진단 모듈을 고려해 볼 수 있다.

III. The Proposed Scheme

1. 모듈 설계

다양한 서비스에서 다뤄지는 데이터와 그 데이터가 실제로 서버에서 다뤄질 때의 메모리와의 관계를 인공지능 모듈에 학습 시키려면 우선 각 언어별 문법과 구문을 학습해야 할 것이고 전문가들이 메모리와 데이터의 관계 패턴 정보를 대량으로 작성하여 기계학습 시켜야 할 것이다. 이와 같은 선례가 없었으므로 확실한 학습을 위해서는 반복된 강화 학습과 지도 학습을 할 필요가 있을 것이다.

날마다 신기술이 나오는 IT 시장의 속도에 모듈의 학습 데이터를 제때 공급하려면 사용자로부터 취약점이나 오류의 발생 부분에 대한 로그를 제공 받아 학습 데이터로 이용하는 것이 중요할 것이라 생각된다.

2. 유의점

인공지능 기술이 발전함에 따라 여러 관련 서비스가 생기는 긍정적인 영향이 많았지만 그 만큼 부정적인 영향도 있었다. 서비스 공격툴 제작이나 악성 코드의 제작에 인공지능 기술이 사용된 사례가 적지 않게 발생하고 있다. 본 논문에서 제안한 진단 모듈이 운영 됐을 때 수집한 데이터들이 역으로 해킹을 위해 사용되지 않도록 보안에 주의를 기울여야 할 것이며, 인공지능 기술을 활용한 해킹으로 인한 피해가 줄어들도록 해당 진단 모듈을 통해 여러 서비스 제공 업체들이 경각심을 지니도록 희망한다.

IV. Conclusions

IT 기술은 정보화 시대답게 점점 빠르게 발전해 나가고 있다. 때문에 제로데이 취약점처럼 모듈이 학습하지 못한 취약점이나 사례가 없던 새로운 취약점이 발생할 가능성이 다분하다.

결국 서비스를 개발하는 사람들 본인이 보안에 대한 경각심을 가지고 이슈에 민감한 자세를 갖출 필요가 있다.

REFERENCES

[1] Log4j - Shell Vulnerability [CVE-2021-44228] “https://nvd.nist.gov/vuln/detail/CVE-2021-44228”

[2] Windows SMB Remote Code Execution Vulnerability (Wannacry) [CVE-2017-0143~4] “https://nvd.nist.gov/vuln/detail/cve-2017-0143“

[3] http://packetstormsecurity.com/files/154690/DOUBLEPULSAR-Payload-Execution-Neutralization.html

[4] https://portal.msrc.microsoft.com/en-US/security-guidance/advisory/CVE-2017-0143