군용 인공지능을 위한 디지털 서명 기반 공급망 보안 체계 구축 방안

장예준* 김록기* 이영화*

*육군미래혁신연구센터

A Signature-Based Supply Chain Security System for Military AI Applications

Yejun Jang* Rocky Kim* Yeonghwa Lee*

*Innovation Institute for Future Army

요 약

군용 인공지능 시스템이 전시 상황에서 해킹되어 무용지물이 되거나, 나아가 적군이 아닌 아군을 겨냥하는 등의 상황은 잠재적으로 치명적이다. 본 연구는 군용 인공지능 시스템을 공급망 취약점을 악용한 변조 공격으로부터 방어하기 위해 디지털 서명 기반 인증 시스템을 구축함을 목적으로 한다. 구체적으로, 국방인증체계(MPKI) 에 사용되는 독자 서명 규격과 호환이 가능하면서도 인공지능의 모델 파일과 훈련 데이터셋 등을 효과적으로 보호할 수 있는 in-toto 공급망 보안 패키지를 활용하여, 군용 인공지능을 위한 공급망 보안 시스템 구축 방안을 제시한다.

I. 서론

1.1 인공지능 보안과 디지털 서명 기술

공격 유형	설명
데이터 포이즈닝	훈련 데이터 변조를 통해 잘못된 예측을 유도하는 공격
모델 포이즈닝	적대자가 머신러닝 모델의 구조나 파라미터를 수정하여 잘못된 동작이나 예측을 유도하는 공격
소프트웨어 포이즈닝	머신러닝 모델을 훈련시키거나 예측을 수행하는 데 사용되는 소프트웨어를 악의적으로 수정하는 공격

표 1 인공지능 공격 유형

인공지능 보안은 인공지능과 연관된 보안 기술 전체를 아우르는 광범위한 용어이다. 본 논문은 포이즈닝 공격(poisoning attack)으로부터 인공지능 시스템을 방어하는 보안 설계에 집중한다. 인공지능 시스템에 대한 변조 공격을 시도할 때 공격자의 주요 타겟은 학습 데이터와 인공지능 모델, 그리고 소프트웨어이다. 이러한 대상에 대한 변조 공격을 각각 데이터 포이즈닝과 모델 포이즈닝, 소프트웨어 포이즈닝이라고 하며, 그 정의는 [표 1]에 나열하였다. Stokes et. al.은 포이즈닝 공격을 “탐지”하는 기존 접근 대신, 올바른 데이터셋과 이로부터 학습된 인공지능 모델을 디지털 서명을 활용하여 인증하는 방안을 제안하였으며, 관련 논문으로 [2]가 있다.

1.2 공급망 보안과 디지털 서명 기술

소프트웨어(SW) 공급망 보안은 소프트웨어 개발주기(SDLC) 전 과정에 걸친 보안을 의미한다.[3] SW 공급망 공격은 오픈소스 소프트웨어의 보안 취약점 및 악성코드를 악용하여 광범위하고 지속적인 피해를 초래하며, 2020년 SolarWinds 사례와 2021년 Log4j 사례가 대표적이다. 개발 생태계 확장으로 네트워크 경계 인증만으로는 공급망 공격 방어가 어렵다는 문제의식이 제로 트러스트(Zero-trust) 패러다임으로 이어졌다. 제로 트러스트 설계 철학이 발전하면서 PKI은 단순 사용자 인증에서의 적용을 넘어 소스코드, 바이너리, 로그 파일 등 개발 과정에서 발생하는 각종 파일들을 서명하고 검증하는 용도로 그 활용성이 확대되고 있다.

1.2 in-toto

in-toto는 퍼듀대학교 교수 Santiago Torres-Arias 주도로 개발되고 있는 전자서명 기반 공급망 보안 시스템으로, GitLab 등 DevSecOps 플랫폼에서 활용되고 있다.

in-toto 기반 공급망 시스템에서 설계자가 각 단계별 보안 사항을 서명해 저장하고, 최종 사용자는 실제로 각 단계가 수행된 결과와 보안 사항을 대조하여 무결성을 검증한다. 구체적으로, 설계자는 각 단계에 사용된 재료(material)와 결과로 생성된 생산물(product)에 대한 정보를 layout 파일에 서명하여 저장하고, 각 단계가 완료될 때마다 실제로 생성된 산출물에 대해 link 파일을 생성해 서명을 남긴다. 최종 사용자는 layout 파일과 각 단계별 link 파일을 대조하여, 공급망 과정에서 변조가 없었음을 확인하고 무결성을 검증한다.

II. 본론

2.1 국방 인공지능 활용 계획

현재 군은 인공지능을 활용하여 GP/GOP, 해강안 경계 등에서 감시정찰체계를 강화하고자 한다. 또한 각군의 C4I 체계를 연동하여 데이터베이스에 산재해 있는 정보의 활용성을 높이고, 전투결심을 비롯한 의사 결정에 지원을 받고자 한다. 국방혁신4.0 기본계획[4]에서는 양질의 국방 데이터셋을 구축하여 인공지능 기반 고성능 무기체계와 전력지원체계의 개발을 기획하고 있다. 군의 무기체계 내에서 인공지능의 판단이 점차 중요해질 것으로 예상되는 만큼, 모델 파일과 훈련 데이터에 대한 무결성을 증명하는 보안 설계는 필수적이다. 특히, “무기체계에서는 주로 변조(무결성)와 거부(가용성)가 보안의 주목표”라는 점[5], 그리고 오픈소스 인공지능 코드베이스에 다양한 취약점이 존재한다는 점을 고려할 때, 적국이 신규 무기체계의 취약점을 역이용하여 전략적 우위를 점하는 시나리오는 충분히 발생 가능하다.

2.2 규격 호환 문제

2.2.1 국방인증체계 (MPKI) 호환 문제

우리 군은 업무 자동화 및 문서 관리, 전자메일 시스템 구축 등을 위해 국방전산망을 구축하여 운영하고 있다. 국방인증체계(MPKI)는 국방전산원을 최상위 인증기관으로 두고 각 군별 등록기관이 사용자 관리를 수행한다.

- Trusted Root CA List

일반적으로, 네트워크의 단말기는 신뢰 가능한 최상위 인증기관 목록(Trusted Root CA List)을 가지고 있어 X.509 체인을 따라 올라가면 해당 목록의 CA가 나오도록 설계되어 있다. 그러나, 국방인증체계는 이와 같은 Known CA list를 활용할 수 없으며, 따라서 대부분의 경우 오픈소스 디지털 서명 라이브러리를 그대로 사용하는데 제약이 있다.

- 독자 서명 알고리즘 및 암호화 알고리즘

신뢰 가능한 인증기관 목록을 MPKI에 알맞게 변형하더라도, KCDSA와 같은 독자 전자서명 규격과 ARIA와 같은 독자 암호 규격을 사용하는 MPKI의 특성 상, 일반적인 X.509 기반 PKI에서 동작하도록 설계된 라이브러리를 사용할 수 없다.

2.2.2 파일 형식 호환 문제

또한, 군의 업무 흐름을 고려할 때 비표준/특이 규격을 취급해야 하는 경우가 다수 존재한다.

- 한글과 컴퓨터 제품군

.hwp, .show, .cell 등 한글과컴퓨터 제품군의 파일 규격에 적용이 가능해야 한다.

- EO/IR/SAR 영상

일반적인 영상 파일과 규격이 다른 원본(RAW) 영상을 처리해야 하는 경우가 다양하게 존재하며, 특히 SAR 영상 처리 시 일반적인 비트맵 형태로 표현되지 않는 .slc 파일을 다루게 될 수 있다.

2.3 in-toto의 국방 활용 가능성

많은 디지털 서명 라이브러리는 국방 분야에서의 활용이 제한된다. 그 이유는 1) Trusted Root CA list가 기 결정되어 있고 2) 사용할 수

있는 서명 알고리즘과 암호화 알고리즘의 목록 역시 결정되어 있으며, 3) application-specific 한 코드에 의해 사용 가능한 파일 포맷의 종류가 제한되기 때문이다. 그러나, in-toto는 다양한 규격과 프로그래밍 언어, 암호 규격에 맞춰 맞춤화가 가능하다. 따라서 군에서 요구하는 자체 규격과 표준에 맞추어 개발할 수 있다. 그러나 in-toto를 군 업무에 활용할 경우, 높은 자유도로 인해 레이아웃 설계를 맡을 보안 전문가의 책임이 커진다는 제약 조건이 따르며, 이에 대한 고려가 필요하다.

2.4 실험 프로그램 작성

본 연구에서는 MNIST 데이터를 활용하여 손글씨 인식 앱을 개발하고 배포하는 전체 과정에 in-toto를 적용하였다. 배시 스크립트를 사용하여 공급망의 모든 과정이 자동으로 실행되도록 코드를 작성하였으며, Ubuntu 20.04 LTS의 Python 3.8 Conda 환경에서 실험을 진행하였다. 자세한 내용은 https://github.com/codingJang/ml-in-toto 에서 확인할 수 있다.

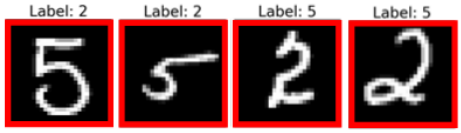

그림 1 위조된 MNIST 데이터셋

데이터 포이즈닝 공격 시나리오를 모델링하기 위해, MNIST 바이너리를 덮어써도 CHECKSUM을 확인하지 않는 torchvision 패키지의 취약점을 발견하고 이를 활용하였다. 구체적으로, run_all.sh에서 --corrupt 플래그를 인자로 받아, 플래그가 False일 경우 원본 MNIST 데이터셋을 사용하여 정상적인 공급망 절차가 진행된다. 반면, 플래그가 True일 경우 [그림 1]와 같이 레이블 2와 5가 서로 뒤바뀐 위조 데이터셋을 기존 데이터셋 위치에 덮어쓰기하여 공급망 내에 위조 데이터셋이 유통되는 시나리오가 진행된다.

[그림 2]에서와 같이, 머신러닝 개발 워크플로우를 가정하였다. Alice는 프로젝트 보안을 담당하며 레이아웃 설계와 데이터셋 제작을 겸한다. 이후, Bob이 CNN 분류기를 훈련하고 Carl이 95% 이상의 정확도로 모델 성능을 평가한 후, Diana가 pyinstaller를 사용해 실행 파일을 생성하여 최종 사용자인 EndUser에게 전달한다. EndUser는 Diana가 빌드한 소프트웨어와 Alice가 작성한 레이아웃 파일, 각 단계별 링크 파일을 통해 공급망의 무결성을 검증한다.

Alice가 작성한 layout 파일에는, EndUser가 앱을 실행할 때 다운로드하여 검증에 사용하는 데이터와 Bob이 학습에 사용한 데이터셋이 일치해야 한다는 MATCH 규칙이 명시되어 있다. Diana는 이 검증을 수행할 in-toto-verify 코드를 최종 배포용 앱에 내장하는 역할을 맡으며, 실제 검증은 EndUser가 앱을 실행할 때 이루어진다. 따라서 정상적인 경우에는 모든 파일이 일치하지만, Bob이 위조된 데이터셋을 사용해 모델을 훈련한 경우 이 규칙을 통과하지 못해 검증에 실패하게 된다.

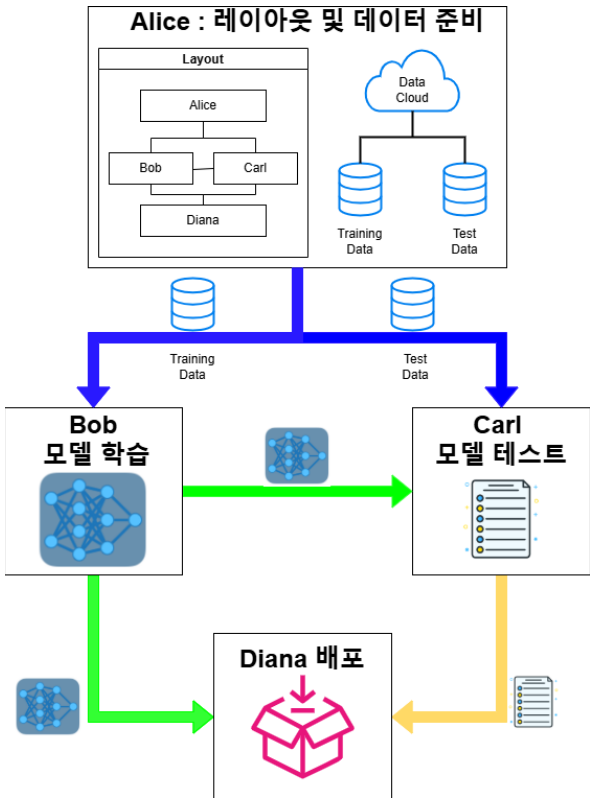

그림 2 머신러닝 개발 워크플로우를 가정함

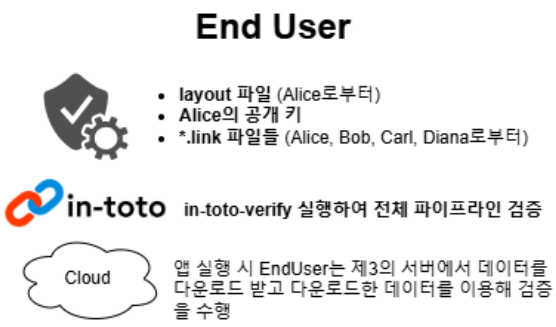

그림 3 EndUser의 무결성 검증

2.5 실험 결과 및 분석

EndUser[그림3]는 각 단계에서 발생한 서명들 사이의 일치 관계를 확인하여 파일의 올바른 전달을 확인하고, MNIST 데이터셋을 믿을 수 있는 제3자로부터 다운받아, 학습에 사용된 데이터셋이 자신이 알고 있는 MNIST 데이터셋과 일치하는지 검증 알고리즘을 통해 확인할 수 있다.

결과적으로, 정상적인 시나리오에서 모든 artifact rule이 통과되면서 앱 상단 바에 “verified app running” 메시지를 출력하였고, 반대로 위조 데이터 셋으로 덮어쓰기 된 시나리오에서 “verification failed” 메시지를 출력하였다. [그림 4]에서 확인할 수 있듯, 검증을 통과한 앱은 숫자 5를 정상적으로 5로 인식하지만, 검증을 통과하지 못한 앱[그림 5]은 5를 2로 인식한다. 또한, 검증 과정에서 서명이 일치하지 않은 파일들을 목록[그림 7]이 정확하게 위조된 데이터셋 파일들과 일치함을 확인할 수 있었으며, 이는 in-toto가 포렌식 과정에 유의미한 도움을 줄 것을 기대해볼 수 있음을 의미한다.

그림 4 검증된 앱 그림 5 검증 실패

```
bash run_all.sh --dry-run
Verifying product rules for 'end-user'...
Verifying 'MATCH data/* WITH PRODUCTS IN mnist-prep/ FROM make-dataset'...
Verifying 'MATCH dist/* WITH PRODUCTS IN mnist-dist/ FROM distribute'...
Verifying 'ALLOW src/download_mnist.py'...
Verifying 'ALLOW alice.pub'...
Verifying 'ALLOW root.layout'...
Verifying 'DISALLOW *'...
The software product passed all verification.
```

그림 6 서명 파일 목록 (검증된 앱)

```
bash run_all.sh --dry-run --corrupt
Queue after 'ALLOW root.layout':
['data/MNIST/raw/train-labels-idx1-ubyte', 'data/MNIST/raw/train-images-idx3-ubyte.gz', 'data/MNIST/raw/t10k-labels-idx1-ubyte.gz', 'data/MNIST/raw/t10k-images-idx3-ubyte.gz', 'data/MNIST/raw/t10k-labels-idx1-ubyte', 'data/MNIST/raw/train-labels-idx1-ubyte.gz']
```

그림 7 서명 불일치 목록 (검증 실패)

III. 결론

본 연구에서는 군용 인공지능 시스템을 공급망 공격으로부터 보호하기 위한 디지털 서명 기반의 인증 체계를 제안하였다. 군에서 사용되는 독자적인 암호화 알고리즘과 인증 체계를 고려하여, in-toto를 활용한 공급망 보안 시스템의 적용 가능성을 검토하였다. 특히, in-toto의 암호화 라이브러리에 대한 맞춤화 용이성과 파일 형식에 구애받지 않는 특성이 군의 MPKI 및 다양한 파일 포맷을 다루는 업무 환경에 적합함을 보였다.

실험을 통해 MNIST 손글씨 인식 프로그램에 in-toto를 적용하여 데이터 포이즈닝 공격을 효과적으로 탐지할 수 있음을 확인하였다. 이는 군사 데이터로 대체될 경우, 감시정찰체계 등에서 인공지능 모델의 무결성을 검증하고 변조된 데이터나 모델의 사용을 방지하는 데 활용될 수 있을 것이다.

향후 연구에서는 군 독자 암호화 및 인증 체계의 in-toto 통합이 필요하다 또한, 보다 복잡한 군용 인공지능 시스템에 대한 적용 사례를 확대하여 실용성을 검증해야 할 것이다. 이를 통해 군용 인공지능 시스템의 보안을 강화하여 안정적 운용 기반을 마련할 것으로 기대된다

[참고문헌]

[1] Stokes, Jack W., Paul England, and Kevin Kane. "Preventing machine learning poisoning attacks using authentication and provenance." MILCOM 2021-2021 IEEE Military Communications Conference (MILCOM). IEEE, 2021.

[2] Jackson, Jon-Nicklaus Z. Exploring Training Provenance for Clues of Data Poisoning in Machine Learning. Diss. 2023.

[3] 한국인터넷진흥원. SW 공급망 보안 가이드라인 1.0, 13 May 2024, 한국인터넷진흥원, https://www.kisa.or.kr.

[4] 국방부. 국방혁신 4.0, 28 Feb. 2023, 대한민국 국방부, https://nsp.nanet.go.kr.

[5] 원경수, 김승주. (2019). 한국군 환경에 적합한 내부자(위협) 정의 및 완화방안 제안. 정보보호학회논문지, 29(5), 1133-1151.