열린 SW 생태계를 위한 과제; 소프트웨어 공급망 보안 강화 필요성과 의의

이만희 한남대학교 컴퓨터공학과 교수

1. 머리말

2020년 12월 미국에서 발생한 솔라윈즈(SolarWinds) 해킹 사건에서, 러시아 정부와 관련된 것으로 알려진 해커들은 먼저 솔라윈즈의 소프트웨어 개발 과정에 침입하여 오리온(Orion) 네트워크 관리 소프트웨어 업데이트에 악성 코드를 추가했다[1]. 이후 이 악성 코드는 솔라윈즈 고객들에게 합법적 소프트웨어 업데이트로 배포되었고, 고객들은 중요한 데이터가 탈취되는 등의 피해를 입었다. 미국 정부 기관, 포춘 500대 기업 및 전 세계의 기관 등을 포함하여 1만 8,000여 기관이 이 해킹에 피해를 입었으며, 피해액은 천문학적인 것으로 알려진다.

솔라윈즈 해킹 사건은 일부 전문가들 사이에서만 논의되던 소프트웨어 공급망 보안을 순식간에 세계적 관심을 받는 중요 보안 주제로 바꾸어 놓았다. 솔라윈즈 해킹으로 큰 피해를 입은 미국은 2021년 5월 대통령 행정명령 14028을 내린다[2]. 미국의 정보보안 수준을 향상시키기 위한 다양한 조치가 실린 이 행정명령은 연방정부의 소프트웨어 공급망 보안 강화를 위해 다각적이고 빠른 실천을 요구하는 일련의 명령을 명시했다. 이 명령에 의해 미국 표준기술연구소(NIST, National Institute of Standards and Technology)와 통신정보관리청(NTIA. National Telecommunications and Information Administration) 등은 미국 정부의 소프트웨어 공급망 보안 강화에 사용될 다양한 규정 및 지침을 작성하였고, 2023년 말까지 모든 연방정부 기관은 이 지침에 의거하여 공급망 보안 관리를 이행해야 한다.

최근 미국은 도입 및 운영 차원에서의 공급망 관리에 그치지 않고, 공급망 보안에서 개발사의 책무를 중요시하는 이른바 'Shift-Left'라고 불리는 국가 사이버 보안 전략(National Cybersecurity Strategy)을 발표하고 관련 제도를 정비하고 있다. 현재 우리나라도 20년 이상 지속되어온 네트워크 경계보안 중점의 사이버 대응 체계에서 벗어나, 내외부 상관없이 아무도 믿지 않는 ZeroTrust를 도입하고 소프트웨어 대한 ZeroTrust 개념인 소프트웨어 공급망 보안 체계로의 전환을 준비하는 등 변화를 추진하고 있다. 본 고에서는 소프트웨어 공급망의 정의, 소프트웨어 공급망 공격의 사례를 통해 살펴보는 공격타입, 미국을 중심으로 한 국내외 정책 변화를 살펴보고, 마지막으로 공급망 보안이 소프트웨어 산업과 교육에 미치는 영향을 살펴본다.

2. 소프트웨어 공급망 소개

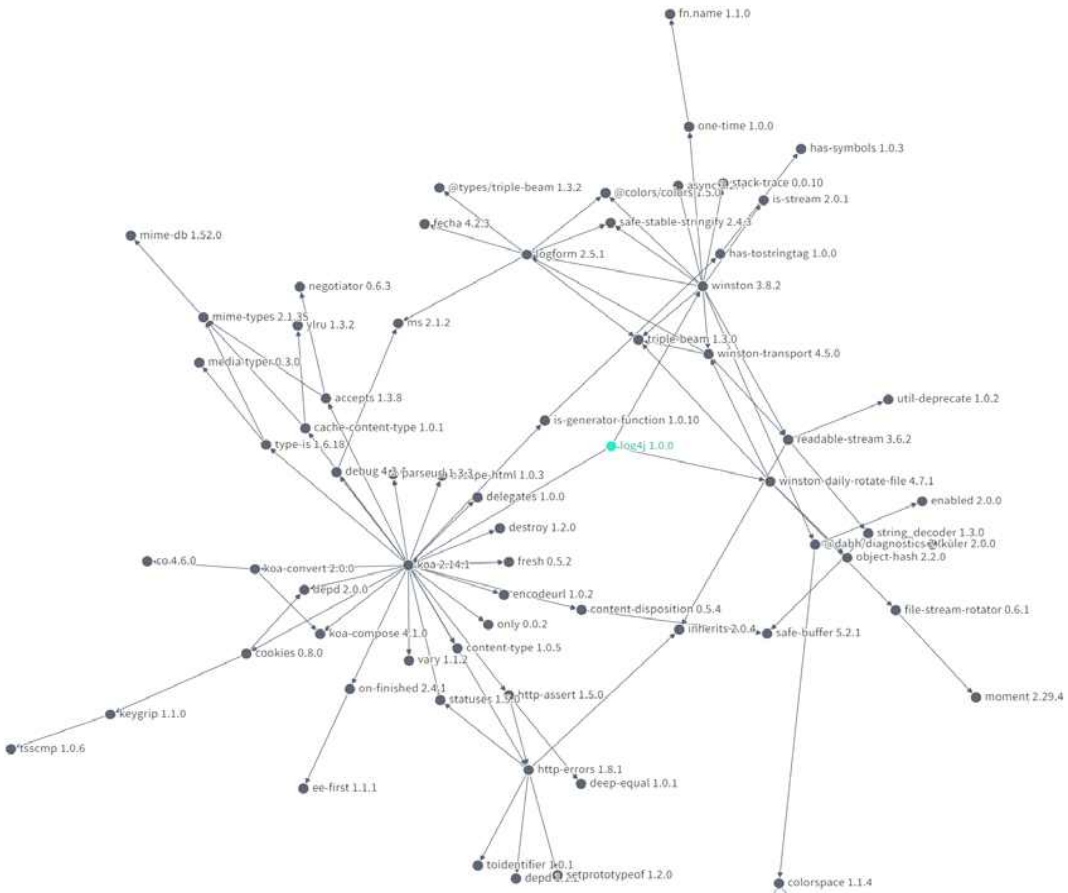
2.1 공급망 정의

공급망이란 공급업체에서 제조업체로 원재료를 전달하는 것부터 최종 사용자에게 최종적으로

전달하는 과정에 관련된 모든 개인, 조직, 자원, 활동 및 기술의 네트워크를 포함한다. 공급망은 제품이나 서비스가 효율적으로 생산되어 고객에게 적시에 전달되는 데 중요한 역할을 한다. 따라서 기업은 효과적인 공급망 관리를 통해 비용을 절감하고 효율성을 높이며, 고객 만족도를 높이기 위해 노력한다. 예를 들어, 기업은 재고관리를 최적화하여 재고 비용을 최소화하고 품절 또는 초과 재고의 위험을 줄일 수 있다. 또한 자동화, 데이터 분석, 인공 지능과 같은 기술을 활용함으로써 공급망 가시성을 개선하고 프로세스를 최적화하며 더 나은 정보에 입각한 의사 결정을 내릴 수 있다.

2.2 소프트웨어 공급망

소프트웨어 공급망은 기존 사업의 공급망 정의와 비슷하게 소프트웨어 제품 및 서비스를 개발, 제공 및 유지 관리하는 프로세스에서 관여되는 모든 개인, 조직, 자원, 활동 및 기술을 의미한다. 과거와 현재의 소프트웨어 공급망을 비교했을 때 가장 두드러진 차이는 코드의 복잡도와 다른 코드의 의존성이다. 과거 소프트웨어는 하드웨어와 소프트웨어가 덜 발달한 시기에 개발되었고 요구 기능 또한 단순했을 뿐 아니라, 외부에서 가져다 사용할 오픈소스 소프트웨어나 라이브러리가 적은 환경이므로 개발자는 모든 것을 직접 개발할 수밖에 없었다. 따라서 과거 소프트웨어는 일반적으로 덜 복잡하고 종속성이 적었다. 그러나 최근 소프트웨어 개발에는 더 빠르고 쉽게 개발할 수 있도록 설계된 여러 라이브러리와 프레임워크, 도구가 사용되기 때문에 소프트웨어의 의존성 또한 매우 복잡해졌다.



[그림1] Log4j 1.0.0 의존성 그래프 [3]

[그림 1]은 2021년 말 세계를 떠들썩하게 한 Log4j 2.x의 초기 버전인 1.0.0의 의존성 정보를 표현한 그림이다[3]. Log4j는 직접적으로는 3개의 패키지인 koa 2.14.1, winston 3.8.2, winston-daily-rotate-file 4.7.1를 의존하지만, 각 패키지는 다시 각각 41개, 29개, 16개의 패키지에 의존하므로 총 73개의 패키지에 의존하고 있다. Google에 의하면 log4j에 영향을 받는 패키지가 3만 5,000개가 넘는다고 하니[4], Log4j 패키지를 사용하는 모든 패키지는 아래 그림과 유사한 공급망을 부가적으로 가진다는 것을 뜻한다. 이러한 공급망의 복잡성으로 인해 의존하는 패키지 하나에 발생한 취약점이 전체 시스템에 큰 파급 효과를 미칠 수 있어, 소프트웨어의 안전한 유지 관리에 큰 어려움이 있다.

2.3 소프트웨어 공급망 공격 유형

소프트웨어 공급망은 소프트웨어 개발 라이프 사이클(SDLC, Software Development Life Cycle)과 뗄 수 없는 관계이다. 소프트웨어 개발 프로세스 자체가 소프트웨어 공급망이기도 하지만, 각 프로세스의 상황을 이용하는 공급망 공격이 일어나기 때문이다. 그러므로 소프트웨어 공급망을 안전하게 관리하기 위해서는 각 소프트웨어 개발 프로세스에서 일어나는 활동과 이에 필요한 자원과 기술이 명세되어야 한다. 본 절에서는 일반적인 소프트웨어 개발 프로세스를 설명하고, 각 단계에서 실제로 발생했던 공격을 예로 들어 소프트웨어 공급망 공격의 유형을 설명한다.

■ 계획 및 설계 단계 공격

소프트웨어 계획 및 설계 단계에서는 이해 관계자와 협력하여 요구 사항을 수집 및 분석하고, 타당성을 평가하고, 소프트웨어 아키텍처 및 기술 구현을 설계하고, 프로토타입을 만들고, 프로젝트 일정 등을 결정한다.

이 단계에서 실제로 발생했던 공격이 2020년 6월 GoldenSpy 공격이다[5]. 한 중국 은행이 외국 기업 고객들에게 설치를 요구한 세무 소프트웨어에 백도어를 삽입한 공격이다. 이는 정상적인 소프트웨어가 사이버 공격에 의해 악성코드화 된 것이 아니라, 소프트웨어 계획 단계에서부터 의도적으로 기획되고 개발된 것으로 알려졌다.

설계 단계 후 개발 프로세스가 시작된다. 최근 소프트웨어 개발 프로세스가 공급망 관점에서 과거와 크게 달라진 점은 SDK(Software Development Kit) 사용, 오픈소스 소프트웨어 사용 증대, 데브옵스(DevOps) 도입 등을 들 수 있다. 이렇게 달라진 요소들로 인해 공급망은 복잡해지고 복잡해진만큼 다양한 공급망 공격에 노출된다. 각 요소별로 실제 발생한 공급망 공격 사례를 살펴본다.

■ 개발단계: SDK 공격

SDK는 특정 플랫폼 또는 장치용 소프트웨어 응용 프로그램을 만드는 데 사용하는 도구, 라이브러리 및 설명서 모음이다. SDK는 개발자에게 플랫폼 또는 장치와 상호 작용할 수 있는 소프트웨어를 구축하는 데 필요한 리소스를 제공한다. 대표적인 SDK로 Android SDK, iOS SDK,

Amazon Web Services SDK 등이 있다.
2020년 7월, 해커들은 클라우드 호스팅 인프라를 통해 CPAAS(Communications Platform as a Service) 회사인 Twilio SDK 라이브러리에 악성 코드를 삽입했다[6]. 이를 통해 해커들은 Twilio SDK 기반 커뮤니케이션 소프트웨어의 민감 정보를 획득할 수 있었다. 즉, 프로그래머가 그 어떤 시큐어 코딩을 적용하더라도, 해당 SDK를 사용한 모든 소프트웨어가 해킹될 수 있음을 뜻한다.

■ 개발단계: 오픈소스 소프트웨어 소스코드 리포지토리 공격
현재 개발자들은 개발 필요 기능 중 가급적 많은 기능을 오픈소스 소프트웨어를 의존하고, 찾을 수 없는 부분만 개발하는 것이 일반적이다. Synopsys의 보고에 의하면, 2018년에는 전체 소프트웨어 코드의 60%가 오픈소스였지만, 2021년에는 이 비율이 78%로 증가했다[7]. 이런 증가세를 고려하면 어떤 소프트웨어의 80% 이상이 오픈소스 소프트웨어라고 해도 과언이 아니다. 그만큼 오픈소스 소프트웨어의 종류와 숫자가 놀랄 정도로 많아졌다.
대부분의 오픈소스 소프트웨어는 소스코드 리포지토리(Source Code Repository)를 이용하여 소스코드를 배포한다. 소스코드 리포지토리는 소프트웨어 개발 과정에서 버전 관리 시스템을 사용하여 코드를 저장하는 곳을 의미한다. 소스코드 리포지토리를 쓰면 개발 팀이 소스코드를 관리하고, 수정, 삭제, 복원 등의 작업을 수행할 수 있으며, 코드 변경을 추적할 수 있어 코드 변경 사항이 서로 충돌할 수 있는 다중 개발자 환경에서는 필수적으로 사용되고 있다. 특히 세계 프로그래머들의 자발적 기여가 중요한 오픈소스 소프트웨어는 반드시 소스코드 리포지토리를 사용할 수 밖에 없고, 대표적인 소스코드 리포지토리로는 GitHub, GitLab, GitBucket 등이 있다.
2021년 3월 대표적인 오픈소스 소프트웨어인 PHP의 공식 Git 서버가 해킹당하여 악성코드가 소스코드에 삽입되는 사건이 발생했다[8]. 삽입된 코드는 PHP 웹서버에 임의의 코드를 실행할 수 있는 치명적 악성코드였다. 다행히 PHP 보안팀에 의해 빠르게 탐지되어, PHP는 모든 PHP 사용자들에게 새로운 PHP로 업데이트하도록 권장했다 하지만 결국 자체 리포지토리 운영을 포기하고, GitHub로 PHP 소스코드를 옮기는 결과를 낳았다[9].

■ 개발단계: 소스코드 리포지토리 타이포스쿼팅 공격
일반적인 타이포스쿼팅(Typosquatting) 공격은 잘 알려진 웹사이트와 유사한 URL을 가진 도메인을 등록하고 진짜 웹사이트와 유사한 웹 사이트를 구축한 후, 사용자의 타이핑 실수로 가짜 웹사이트를 방문했을 때 사용자의 개인정보나 금융정보를 탈취하는 공격이다. 최근 증가하고 있는 리포지토리에 대한 타이포스쿼팅은 Dependency Confusion 공격이라도 불리며, NPM (Node Package Manager) 또는 PyPI(The Python Package Index)와 같은 공개 소스 코드 리포지토리의 적법한 패키지 또는 라이브러리와 매우 유사한 이름으로 패키지 또는 라이브러리를 등록한다. 이후 개발자가 실수로 해당 가짜 패키지 이름을 입력하면, 이를 통해 맬웨어, 백도어 또는 기타 유해한 코드를 최종 소프트웨어 탑재하는 것을 목표로 한다.
2022년 2월 GitHub에 Rust의 통화 변환, 세금 계산 및 회계 소프트웨어와 같이 정확한 소수점

계산이 필요한 금융 및 통화 응용 프로그램에 주로 사용되는 패키지인 rust_decimal과 소스코드가 거의 똑같지만, Decimal::new 함수에서 악성코드를 내려받는 부분이 다른 rustdecimal이 등록되었다[10].
이와 유사한 공격으로 2021년 2월 Alex Birsan이라는 연구자가 Apple, Microsoft 등 주요 회사를 공격한 방법이 있다. 회사 내부 private 소스코드 리포지토리에 존재하는 패키지와 이름과 똑같은 패키지를 public 소스코드 리포지토리에 등록, private 패키지보다 먼저 최종 소프트웨어에 사용될 수 있음을 보였다[11].

■ 개발단계: 빌드 시스템 공격
또 하나의 주요 개발 트렌드인 DevOps는 소프트웨어 개발(Dev)과 IT 운영(Ops)을 결합하여 소프트웨어 개발 속도, 안정성 및 품질을 개선하는 소프트웨어 개발 방법론이다. DevOps는 협업과 자동화를 촉진하여 개발 팀과 운영 팀 간 장벽을 허무는 것을 목표로 한다. 이를 위해 사용하는 전략이 CI/CD(Continuous Integration/Continuous Delivery)이다. 수정된 코드에 대해 자동 빌드·자동 테스트를 수행하고, 최종 서비스 또는 제품까지 자동으로 배포되는 시스템을 구축하여 개발팀과 운영팀이 협업한다.
Jenkins는 가장 많이 사용되는 CI/CD 도구 중 하나다[12]. Jenkins는 개발자의 코드 변경과 같은 이벤트에 의해 트리거 되거나 특정 시간에 예약될 수 있는 자동화된 작업을 정의하여, 코드 테스트 실행이나 코드 빌드 및 패키징, 서버 또는 클라우드 플랫폼에 대한 애플리케이션 배포와 같은 다양한 작업을 자동으로 수행할 수 있다.
2023년 3월 Jenkins 서버에 임의의 실행코드를 실행할 수 있는 심각한 취약점이 발견되었다[13]. 이렇게 빌드 시스템이 해킹되면 발생할 수 있는 대표적인 사건이 솔라윈즈 공격이다. 해커들은 솔라윈즈의 오리온 소프트웨어 개발에 사용되는 빌드 시스템을 해킹하여 최종 소프트웨어 Sunburst 백도어를 삽입하였고, 이 코드는 업데이트 서버로 전송된 후, 소프트웨어 업데이트를 수행한 모든 시스템을 감염시켰다[1].

■ 테스팅 단계: 테스팅 환경 공격
CI/CD에서는 자동 테스팅이 필수적이다. Codecov는 이를 대신해주는 서비스이다. 소프트웨어 코드의 품질과 보안을 개선할 수 있도록 소프트웨어 테스트 및 코드 커버리지 분석 도구와 서비스를 제공한다[14]. 2021년 4월 발생한 공격의 경우, 도커(Docker) 이미지의 취약점을 활용해 배시 업로더(Bash Uploader) 스크립트에 접근하여 사용자의 GitHub 액세스 토큰을 포함하여 Codecov 고객의 자격 증명을 획득하였고, 이를 사용하여 GitHub에서 고객의 소스 코드 리포지토리에 대한 무단 액세스 권한을 얻었다[15].

3. 국내외 공급망 보안 정책

3.1 미국 공급망 보안 정책

3.1.1 미국 대통령 행정명령 Executive Order(EO) 14028 바이든 행정부가 발표한 EO 14028은 "Improving the Nation's Cybersecurity"라는 제목으로 2021년 5월 21일 발표되었으며[2], 특히

섹션 4는 소프트웨어 공급망 보안 강화에 중점을 두었다. 이 섹션에서는 연방 정부가 사용하는 소프트웨어의 보안이 매우 중요하며, 소프트웨어가 의도한 대로 안전하게 작동하도록 보장하기 위해 보다 엄격하고 예측 가능한 메커니즘이 필요함을 강조하였다. 이 문제를 해결하기 위해 NTIA와 NIST에 연방 정부의 공급망 보안 향상을 위한 지침 작성을 명령하였고, 다음은 지침서의 주요 내용이다.

■ Critical Software

먼저 보호대상 소프트웨어를 지정하기 위해 Critical Software를 정의하였다[16]. 일명 EO-Critical Software는 다음 중 하나 이상의 특징을 가진 구성 요소와 직접적인 소프트웨어 의존성이 있는 소프트웨어로 정의된다.

- 권한을 관리하거나 상승된 권한을 가지고 실행되도록 설계된 소프트웨어
- 네트워크 또는 컴퓨팅 자원에 직접 또는 특권적인 액세스가 있는 소프트웨어
- 데이터 또는 운영 기술에 대한 액세스를 제어하는 소프트웨어
- 신뢰에 대한 중요한 역할을 수행하는 소프트웨어
- 특권적인 액세스와 함께 일반적인 신뢰 경계 외부에서 작동하는 소프트웨어

■ Secure Software Development Framework & Minimum Standards for Developer Verification of Software

소프트웨어 개발을 위한 지침으로는 Secure Software Development Framework (SSDF, SP 800-218)과 Guidelines on Minimum Standards for Developer Verification of Software (NISTIR 8397)이 개발되었다[17][18]. SP 800-218은 소프트웨어 보안을 개발 초기 단계에서부터 고려하는 것이 중요하다는 것을 강조하며, 보안 요구 사항을 정의하고, 보안 위험을 식별하고, 적절한 보안 제어를 선택하고, 보안 기능을 테스트하여 소프트웨어 보안성을 보장하는 프레임워크이다. 이 프레임워크는 다양한 보안 기술과 방법을 적용하여 소프트웨어 보안을 강화하는 것을 목적으로 하며, 기업이나 조직에서 적용하여 소프트웨어 보안성을 개선하는 데 도움을 준다.

NISTIR 8397은 소프트웨어 개발자가 소프트웨어 제품을 검증하기 위한 최소 기준을 제공한다. 이 문서에서는 소프트웨어 개발자가 자신들의 제품을 안전하게 사용하기 위해 어떤 검증 기능을 사용할 수 있는지, 이러한 검증 절차가 소프트웨어 보안에 어떤 영향을 미치는지, 또한 이러한 검증 절차가 제품 품질 향상에 어떤 역할을 하는지 등에 대해 다룬다. 이러한 검증 기능에는 소프트웨어 보안 요구사항을 준수하는지 확인하기 위한 정적 및 동적 분석, 코드 리뷰, 사용자 입력 검증, 테스트 등이 포함된다. 이러한 기능들은 소프트웨어 개발 초기 단계부터 적용되어야 하며, 소프트웨어 개발 수명주기 동안 지속적으로 수행되어야 한다.

■ Minimum Elements for SBOM

대통령행정명령은 미국 상무부의 NTIA를 통해 SBOM(Software Bill of Material)의 최소 요를 개발했고, 포맷으로는 SPDX, CycloneDX, SWID를 사용하도록 정했다[19].

■ Cybersecurity Supply Chain Risk Management Practices for Systems and Organizations

대통령 행정명령은 NIST SP 800-161의 수정을 통해 소프트웨어를 도입하는 각 조직이 전체 공급망에서 사이버 보안 위험을 해결하기 위한 프레임워크를 제공, 공급망 위험관리 프로세스를 향상시키기 위한 지침을 제공한다[20]. 이 문서는 각 조직이 공급망 내에서 개발, 통합, 배포 과정에서 사용된 프로세스, 절차, 표준 등이 어떻게 사용되었는지에 대한 가시성과 이해력이 부족하여 다양한 공급망 위험이 존재하고 있음을 지적한다. 이 문서는 이러한 위험은 기업의 기술 획득 방식 및 조직 내 모든 수준에서 사이버 보안 공급망 리스크 관리(C-SCRM, Cybersecurity Supply Chain Risk Management)를 적용함으로써 식별, 평가 및 완화될 수 있다고 밝힌다. 또 C-SCRM 전략 실행 계획, C-SCRM 정책, C-SCRM 계획 및 제품 및 서비스 위험 평가를 포함한 다중 수준의 C-SCRM 특정 접근 방식을 적용하여 C-SCRM을 위험 관리 활동에 통합하는 가이드를 제공한다. 전반적으로 NIST SP 800-161은 조직이 사이버 보안 상태를 개선하고, 공급망을 잠재적인 사이버 보안 위협으로부터 보호할 수 있도록 설계되었다.

3.1.2 EO-14028 실행 타임 라인 공표

대통령 행정명령이 내려진 지 약 1년 6개월 후인 2022년 9월 14일, 미국 예산 관리국(OMB, Office of Management and Budget)은 "Enhancing the Security of the Software Supply Chain through Secure Software Development Practices"(M-22-18)을 통해 연방 정부 기관들이 공급망 보안관리를 수행해야 할 주요 업무와 수행 날짜를 지정하였다[21]. 각 연방 기관은 소프트웨어를 사용하기 전 소프트웨어 개발사로부터 자체증명서(Self-attestation)를 획득해야 한다. 이 자체증명서는 개발사가 NIST 규정을 잘 준수했다는 셀프 증명서이며, 개발사 이름과 제품 설명, Secure Software Development를 잘 따랐는지 증명하는 내용 등이 포함된다. 필요에 따라 제3자 보증도 포함될 수 있다. 또한 기관은 필요할 경우 소프트웨어 개발사로부터 SBOM을 포함한 다양한 결과물을 요구할 수 있다. 주요 타임라인은 <표 2>와 같다.

<표 1> The Minimum Elements For a Software Bill of Materials(SBOM)

항목명	항목 설명
공급자 이름	구성 요소를 생성, 정의 및 식별하는 개체의 이름
구성 요소 이름	원래 공급자가 정의한 소프트웨어 단위에 지정된 이름
구성 요소의 버전	이전에 식별된 버전에서 소프트웨어의 변경을 지정하는 데 공급자가 사용하는 식별자
기타 고유 식별자	구성 요소를 식별하는 데 사용되거나 관련 데이터베이스에서 참조 키로 사용되는 기타 식별자
의존성 관계	상위 구성 요소 X가 소프트웨어 Y에 포함된 관계
SBOM 데이터의 작성자	이 구성 요소에 대한 SBOM 데이터를 작성하는 개체의 이름
타임스탬프	SBOM 데이터 어셈블리 날짜와 시간의 기록

<표 2> 연방 정부 기관의 공급망 보안 관리를 위한 주요 타임라인

기한	수행 내용
공표 후 90일까지(22.12.13)	Critical SW 목록 작성
공표 120일까지(23.1.12)	SW 개발사와 소통할 수 있는 일관된 프로세스를 수립하고, 공개되지 않은 자체증명서는 하나의 관리 시스템을 통해 수집되도록 준비
공표 270일까지(23.6.11)	Critical SW의 자체증명서를 받아야 함
공표 365일까지(23.9.14)	모든 SW의 자체증명서를 받아야 함

3.1.3 미국 국가 사이버 보안 전략

2023년 3월 2일 바이든 대통령은 국가 사이버 보안 전략를 발표하였다[22]. 주요 내용은 주요 기반시설 보호 강화, 위협 행위자 교란 및 해체, 보안 및 복원력을 촉진하기 위한 시장 변화, 회복력 있는 미래를 위한 투자, 공동 목표를 추구하기 위한 국제적 파트너십 구축이다. 특히 시장 변화에 대한 부분에서는 일명 'Shift-Left'로 불리는 움직임이 뚜렷하다. 즉, 보안의 책무를 사용자나 정부에서 개발사로 이동하는 것이다. 3.5절 "Shift Liability for Insecure Software Products and Services"에서는 안전한 소프트웨어 개발을 위해 적절한 조치를 취하지 않은 기업은 이에 대한 책임을 져야 함을 밝혔으며, 향후 이 책임을 명확하게 하기 위한 제도 도입을 위해 국회와 협업할 예정이다. 이와 함께 safe harbor framework 개발을 통해 적절한 사이버 보안 성숙도를 증명하는 민간 기업에 대한 책임 면제 보호를 제공할 것이다. 이는 보안 사고가 생겼을 때 책임에 대한 면책 요건이 되기 때문에 민간 기업에게는 상당한 인센티브가 될 것이다. 따라서 많은 기업이 보다 적극적으로 이 자격을 얻기 위해 노력할 것으로 예상된다. 이 자격에 대한 자세한 규정은 공개되지 않은 상태이다.

3.2 우리나라 공급망 보안 정책

우리나라는 2019년 9월 부처 합동으로 "국가사이버안보 기본계획"의 18개 중점과제 중 하나로 공급망 과제를 선정, 주요 정보통신 기반 시설의 안정적 운영을 위한 공급망 보안 체계 구축과 주요 공공기관에 도입되는 ICT 장비의 공급망 보안 관리 체계 구축을 목표로 설정하였다[23]. 하지만 실제적인 체계 구축은 발표되지 않은 채, 과기정통부 주도로 2021년 2월 "K-사이버방역추진전략"을 발표하였고, 2023년까지 '공공기관에 도입되는 SW의 안정성 점검 및 공급망 관리도구 보급', '주요 SW 개발 및 서비스 기업에 대한 개발 환경 보안점검 지원 및 인증 제도 도입'을 주요 목표로 정했다[24]. 이와 관련된 일부 연구가 진행되고는 있으나, 현재까지 공공기관에 보급되는 공급망 관리 도구나 개발환경 점검 지원 및 인증제도는 준비되지 않은 실정이다. 과기정통부는 2022년 10월, 약 40여명의 전문가로 구성된 "제로트러스트 & 공급망 보안 포럼"을 발족하였다[25]. 산업·정책 분과는 공급망 보안 향상을 위한 다양한 정책을 개발하고 있으며, 기술·표준 분과는 ICT 장비 공급망 점검 방안을 검토하고 있다.

한편, 국가정보원은 보안적합성 검증 제도를 운영하고 있다. 국가정보통신망의 보안수준 제고를 위해「국가정보원법」제4조와「전자정부법」제56조에 의거, 국가·공공기관이 도입하는 정보보호시스템·네트워크 장비 및 양자암호통신장비 등 보안기능이 탑재된 IT제품 및 저장자료 완전삭제제품의 안전성을 검증하는 제도이다[26]. 2022년 10월 발표된 신 보안적합성 검증체계

에서 국제 및 국가배후 해킹조직이 개발·유포에 관여한 IT보안제품의 공공분야 유입을 최대한 억제하고 공급망 보안을 강화하기 위해 각급기관과 보안업계의 주의환기를 요청하였고, IT보안제품 도입시 우리나라를 비롯, 국제사회의 제재를 받는 국가(단체) 연루 여부 확인을 필수화하고 있다[27]. 수정된 검증체계로 인해 제재국으로부터의 도입은 막을 수 있지만, 현재의 보안적합성 검증제도로 공급망 보안을 관리하기에는 한계가 있다. 첫째, 보안적합성 검증제도는 IT보안제품에 한정되므로 일반 SW에 대한 공급망 관리에 한계가 존재한다. 둘째, 소프트웨어에 사용되고 있는 다양한 오픈소스는 불특정 다수 개발자의 코드 기여로 이루어지므로 국가를 특정하기 어려운 상황이라, 오픈소스의 국가 출처 확인을 통한 제재국 식별은 해결하기 쉬운 문제가 아니다.

4. 소프트웨어 공급망 보안이 소프트웨어 산업에 미치는 영향

소프트웨어 공급망 보안의 필요성이 증대되고 관련된 규정이 제정되면 소프트웨어 산업 전반에 큰 영향을 미칠 것으로 예상된다. 긍정적인 면으로는 소프트웨어 공급망 보안에 대한 관심과 투자 증대로 소프트웨어 제품의 전반적인 품질과 신뢰성 향상에 도움이 되고, 이는 소프트웨어 개발에 있어 취약점을 줄이는 데 도움이 될 것이다. 또한 보안성 제공이 중요한 산업의 경우, 경쟁우위 확보를 위한 적극적 투자가 이루어질 것이고 이는 기업의 주요 강점이 될 수 있다. 또한 소프트웨어 공급망 신뢰성 확보라는 새로운 보안 분야의 생성으로 인해, 관련된 도구와 프로세스의 개발은 물론, 이와 관련된 컨설팅 기업 및 스타업이 등장하는 기회가 될 수 있다.

그러나 소프트웨어 공급망 보안의 중요성 증대는 소프트웨어 산업에 부정적인 영향도 미칠 수 있다. 안전한 소프트웨어 공급망을 구현하기 위해 시간과 비용을 추가 투자해야 하고, 이 때문에 소프트웨어 개발 비용이 증대될 수 밖에 없다. 자연히 동일한 기능을 하는 소프트웨어를 더 비싼 가격으로 구매해야 한다. 단기적으로는 관련 분야 전문가가 부족하므로, 이미 개발 인력 수급에 어려움을 겪고 있는 중소기업 및 스타업은 공급망보안 관련 요구사항을 만족시키기 어려운 반면, 비교적 인력 충원이 유리한 대기업에게는 오히려 사업적 기회가 될 수 있어 소프트웨어 산업의 불평등 구조를 악화시킬 수 있다. 또한 빠른 혁신이 필요한 초경쟁사회에서 새로운 기능과 서비스를 개발하고 그 효용성을 검증하는데 사용할 시간을 공급망 보안을 위해 사용해야 하기 때문에 창의적 기업 프로세스에 걸림돌이 될 수 있다.

하지만 이러한 어려움 때문에 공급망 보안에 대한 적극적인 대처를 하지 않는다면, 새로운 취약점과 공급망 공격에 속수무책일 수밖에 없다. 산업이 스스로 문제를 해결할 수 없을 때는 정부가 해결의 실마리를 제공할 수 있다. 정부는 다양한 접근법을 통해 소프트웨어 산업의 전환을 도울 수 있다. 첫째, 공급망 보안 개발과 관련된 자동화된 도구를 국가 기관 주도로 개발한 후, 이를 적극적으로 보급할 수 있다. 둘째, 공급망 보안과 관련된 기술 지원을 담당하는 공급망 보안 기술지원 센터를 지역별로 운영할 수 있다. 이 기술 지원 센터는 예산과 인력을 확보하여 관련 기술 개발 및 재직자 교육 등을 상시적으로 수행할 수 있다. 셋째, 중소기업 및 공공 기관에 대한 개발 자금 지원 또는 세제 혜택을 통해 공급망 보안에 보다 적극적인 투자를 유도할 수 있다. 넷째, 창업지원 인큐베이팅 과정에 공급망 보안을 포함한 다양한 정보보안 관

련 컴플라이언스 이슈를 지원하는 프로세스를 구축하여 창업하는 기업은 제품 자체에 집중할 수 있도록 도울 수 있다.

공급망 보안을 포함한 정보보안 문제는 정보보안 산업의 문제에서 벗어나 개발자와 개발자를 양성하는 대학 교육의 변화로 이어져야 한다. 행정기관 및 공공기관 정보시스템 구축·운영 지침은「전자정부법」제45조제3항에 따라 행정기관 등의 장이 정보시스템을 구축·운영함에 있어서 준수해야 할 기준, 표준 및 절차와 법 제49조제1항에 따른 상호운용성 기술평가에 관한 사항을 정하고 있다. 이 지침에 의하면 소프트웨어 보안 약점이 없도록 소프트웨어 개발 보안을 적용하도록 하고 있다. 즉, 공공기관에 도입되는 소프트웨어를 개발하기 위해서는 소프트웨어를 안전하게 개발하는 능력이 필요하다.

하지만 개발자를 양성하는 국내 대학 컴퓨터공학과 및 소프트웨어 관련 학과들은 정보보안 관련 과목들을 운영하고는 있지만, 소프트웨어 보안이나 시큐어코딩 과목을 운영하는 학과는 매우 드물다. 산업의 요구에 민감하게 반응하는 현대학 운영 상황에서 시큐어코딩 과목 운영이 활성화되지 않는 이유는 현재 산업에서 시큐어코딩 능력을 갖춘 인재가 꼭 필요하지 않기 때문이다. 현재 정보처리기사 교과의 실기과목에 "소프트웨어 개발 보안"이 있지만, 개발 보안에 대한 이론을 다루므로, 대학 현장에서 실제 프로그래밍에서의 시큐어코딩을 가르칠 필요성을 느끼지 못한다[28]. 이러한 상황에서 대학에서 공급망 보안에 관련된 교육을 기대하는 것은 무리다.

이를 타개하기 위해서는 다시 정부의 역할이 필요하다. 교육부 또는 과기정통부는 시큐어코딩 및 공급망 보안 관련 과목을 운영하는 전체 학과들이 혜택을 얻을 수 있는 사업을 기획할 수 있다. 이와 함께, 관련 온라인 콘텐츠를 제작하고 온라인 교육 수료증 등을 발급하여, 이를 졸업학점에 반영하거나, 해당 교육을 수강한 재직자라야 국가 기관 소프트웨어 개발에 참여하게 하는 등의 규정 개정을 통해 국가·공공 기관에 도입되는 소프트웨어 보안 품질을 높이는 동시에, 대학이 공급망 보안을 포함한 소프트웨어 개발 보안 교육에 더 투자할 수 있도록 장려할 수 있다.

5. 맺음말

솔라윈즈 사건으로 시작된 공급망 보안 이슈는 바이든 정부의 행정명령 14028로 인해 더욱 주목받고 있고, 최근 국가 사이버 보안 전략에도 중요 내용으로 언급되고 있다. 그런데 공급망 보안은 다른 일반 보안 문제와 다르게 다루어지고 있다는 것이 중요하다. 일반 사이버 보안 문제의 경우, 주로 보안 기술 개발을 통한 사이버 공격 탐지 및 대응 능력 향상과 함께 관련 보안 문제를 해결하는 기관 지정 및 인력 보강 등 사이버 보안 영역 내에서 대응이 이루어졌다. 하지만 소프트웨어 공급망 보안 분야는 식별이 불가능할 정도로 많은 조직과 개인이 소프트웨어 개발에 연관되어 있고, 따라서 공급망의 폭과 깊이만큼 attack surface가 넓어, 이제까지의 보안 문제처럼 소프트웨어 사용자가 공급망 공격 전체를 대응하는 것은 불가능하다. 이 때문에 미국도 개발사의 책임을 더욱 중요시하는 Shift-Left 전략을 취하고 있다. 공급망 보안과 관련하여 적절한 조치를 취하는 기업에게는 면책 혜택을, 그렇지 않은 기업에게는 책임을 추궁하는 전략으로 선회하고 있는 것이다.

이렇게 증가하는 소프트웨어 공급망 보안의 중요성은 소프트웨어 산업에 영향을 미칠 수밖에 없고, 그 영향은 긍·부정의 면을 모두 가질 것으로 예상된다. 소프트웨어 공급망 보안에 대한 투자는 소프트웨어 제품의 품질과 신뢰성을 향상시키고 새로운 비즈니스 기회를 제공하며 경쟁 우위를 창출할 수 있다. 반면, 소프트웨어 개발 비용을 증가시켜 중소기업 및 신생 기업이 보안 요구 사항을 준수하기 어렵게 만들고, 혁신을 방해할 수도 있다. 이에 정부는 자동화 도구 개발, 기술 지원 센터 운영, 세금 인센티브 등 다양한 방법으로 소프트웨어 산업 변화를 유도할 수 있다. 또한 안전한 코딩 능력을 갖춘 개발자를 양성하기 위한 대학 교육의 변화를 유도할 수 있다. 이를 위해 정부는 공급망 보안을 포함한 소프트웨어 개발 보안 교육을 확대하고, 관련 온라인 콘텐츠 제작, 온라인 교육 인증서 발급, 소프트웨어 개발 보안 교육 관련 규정 개정을 통해 소프트웨어 개발 보안 교육을 장려할 수 있다.

이 기고문을 작성하는 2023년 4월 초 현재도 북한발 공급망 보안 공격으로 인해 '이니세이프'의 취약점이 악용되어 공공기관, 방산, 바이오 기업 등이 피해를 입었고[29], 기업 커뮤니케이션앱인 3CX의 빌드 시스템이 점거되고 자동으로 업데이트되면서 해당 앱을 사용하는 기업이 피해를 입는 전형적 공급망 공격이 발생하고 있다[30]. 가까운 미래에 안전, 교통, 의료, 에너지, 유통 등 사회 전 분야에서 소프트웨어에 대한 의존성이 지금보다 훨씬 더 높아질 것은 자명하다. 이런 환경에서 발생하는 북한발 또는 국제 해킹 그룹의 공급망 공격은 국민의 안전과 건강, 사회 안전 유지에 큰 위협이 될 국가적 안보 이슈라 할 수 있다. 지금 바로 움직여야 한다.

※ 본 연구는 2023년도 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(NRF-2021R1A4A2001810).

[참고문헌]

[1] New York State Department of Financial Services, Report on the SolarWinds Cyber Espionage Attack and Institutions' Response, https://www.dfs.ny.gov/system/files/documents/2021/04/SolarWinds_report_2021.pdf

[2] The White House, Executive Order on Improving the Nation's Cybersecurity, https://www.whitehouse.gov/briefingroom/presidential-actions/2021/05/12/executive-order-on-improving-the-nations-cybersecurity/

[3] NPM package log4j dependencies, https://deps.dev/npm/log4j/1.0.0/dependencies/graph

[4] Recorded Future News, Google: More than 35,000 Java packages impacted by Log4j Vulnerabilities, https://therecord.media/google-more-than-35000-java-packages-impacted-by-log4j-vulnerabilities

[5] DARKReading, 'GoldenSpy' Malware Hidden in Tax Software Spies on Companies Doing Business in China, https://www.darkreading.com/threat-intelligence/-goldenspy-malware-hidden-in-tax-software-spies-on-companies-doingbusiness-in-china

[6] Twilio, Incident Report: TaskRouter JS SDK Security Incident - July 19, 2020, https://www.twilio.com/blog/incidentreport-taskrouter-js-sdk-july-2020
[7] Synopysis, 2022 OPEN SOURCE SECURITY AND RISK ANALYSIS REPORT
[8] BLEEPINGCOMPUTER, PHP's Git server hacked to add backdoors to PHP source code, https://www.bleepingcomputer.com/news/security/phps-git-server-hacked-to-add-backdoors-to-php-source-code/
[9] PHP.WATCH, git.php.net server compromised, move to GitHub, and delayed updates, https://php.watch/news/2021/03/git-php-net-hack
[10] Cycode, TypoSquatting, RepoJacking, and Domain Takeover – The Story of the Recent Attacks, https://cycode.com/typosquatting-repojacking-domain-takeover/
[11] Medium.com, Dependency Confusion: How I Hacked Into Apple, Microsoft and Dozens of Other Companies, https://medium.com/@alex.birsan/dependency-confusion-4a5d60fec610
[12] Jenkins, https://www.jenkins.io/
[13] Aqua Blog, CorePlague: Severe Vulnerabilities in Jenkins Server Lead to RCE, https://blog.aquasec.com/jenkins-servervulnerabilities
[14] CodeCov, https://about.codecov.io/
[15] 보안뉴스, 코드 점검 서비스 회사 코드코브, 수개월 동안 침해 사실 몰랐다, https://www.boannews.com/media/view.asp?idx=96653
[16] NISTI, Critical Software Definition, https://www.nist.gov/itl/executive-order-improving-nations-cybersecurity/criticalsoftware-definition
[17] NIST, SP 800-218, Secure Software Development Framework (SSDF) Version 1.1: Recommendations for Mitigating the Risk of Software Vulnerabilities, https://csrc.nist.gov/publications/detail/sp/800-218/final
[18] NIST, NISTIR 8397, Guidelines on Minimum Standards for Developer Verification of Software, https://csrc.nist.gov/publications/detail/nistir/8397/final
[19] NTIA, The Minimum Elements For a Software Bill of Materials (SBOM), https://www.ntia.gov/report/2021/minimumelements-software-bill-materials-sbom
[20] NIST, SP 800-161 Rev. 1, Cybersecurity Supply Chain Risk Management Practices for Systems and Organizations, https://csrc.nist.gov/publications/detail/sp/800-161/rev-1/final
[21] Office of Management and Budget, Enhancing the Security of the Software Supply Chain to Deliver a Secure Government Experience, https://www.whitehouse.gov/omb/briefing-room/2022/09/14/enhancing-the-security-of-the software-supply-chain-to-deliver-a-secure-government-experience/
[22] The White House, FACT SHEET: Biden-Harris Administration Announces National Cybersecurity Strategy, https://www.whitehouse.gov/briefing-room/statements-releases/2023/03/02/fact-sheet-biden-harris-administration-announcesnational-cybersecurity-strategy/

[23] 관계부처 합동, 국가 사이버안보 강화를 위한 이행방안 확정,
https://www.mois.go.kr/cmm/fms/FileDown.do?atchFileId=FILE_00088335yTEEjB_&fileSn=0
[24] 관계부처 합동, K-사이버방역 추진 전략,
https://www.korea.kr/common/download.do?fileId=193918080&tblKey=GMN
[25] 과학기술정보통신부, 제로트러스트 공급망 보안 정책포럼 발족식,
https://www.msit.go.kr/bbs/view.do;jsessionid=XPwj-hFU_TCshCvmPhzWTyBlU6MPCUsErRmRW__1.AP_msit_2?sCode=user&nttSeqNo=3176901&bbsSeqNo=87&mId=116&mPid=111
[26] 국가정보원, 보안적합성 검증, https://www.nis.go.kr:4016/AF/1_7_2_1.do
[27] 국가정보원, 공공분야 도입운용 IT보안제품 新 보안적합성 검증체계,
https://www.nis.go.kr:4016/resources/synap/skin/doc.html?fn=NIS_FILE_1664932468164
[28] 한국산업인력관리공단, 정보처리기사,
https://www.q-net.or.kr/crf005.do?id=crf00503&jmCd=1320
[29] 국가정보원, 이니텍社 '이니세이프' 최신 보안업데이트 권고,
https://www.ncsc.go.kr:4018/main/cop/bbs/selectBoardArticle.do?bbsId=SecurityAdvice_main&nttId=32172&pageIndex=1
[30] ReversingLabs, Red flags flew over software supply chain-compromised 3CX update,
https://www.reversinglabs.com/blog/red-flags-fly-over-supply-chain-compromised-3cx-update

※ 출처: TTA 저널 제206호