



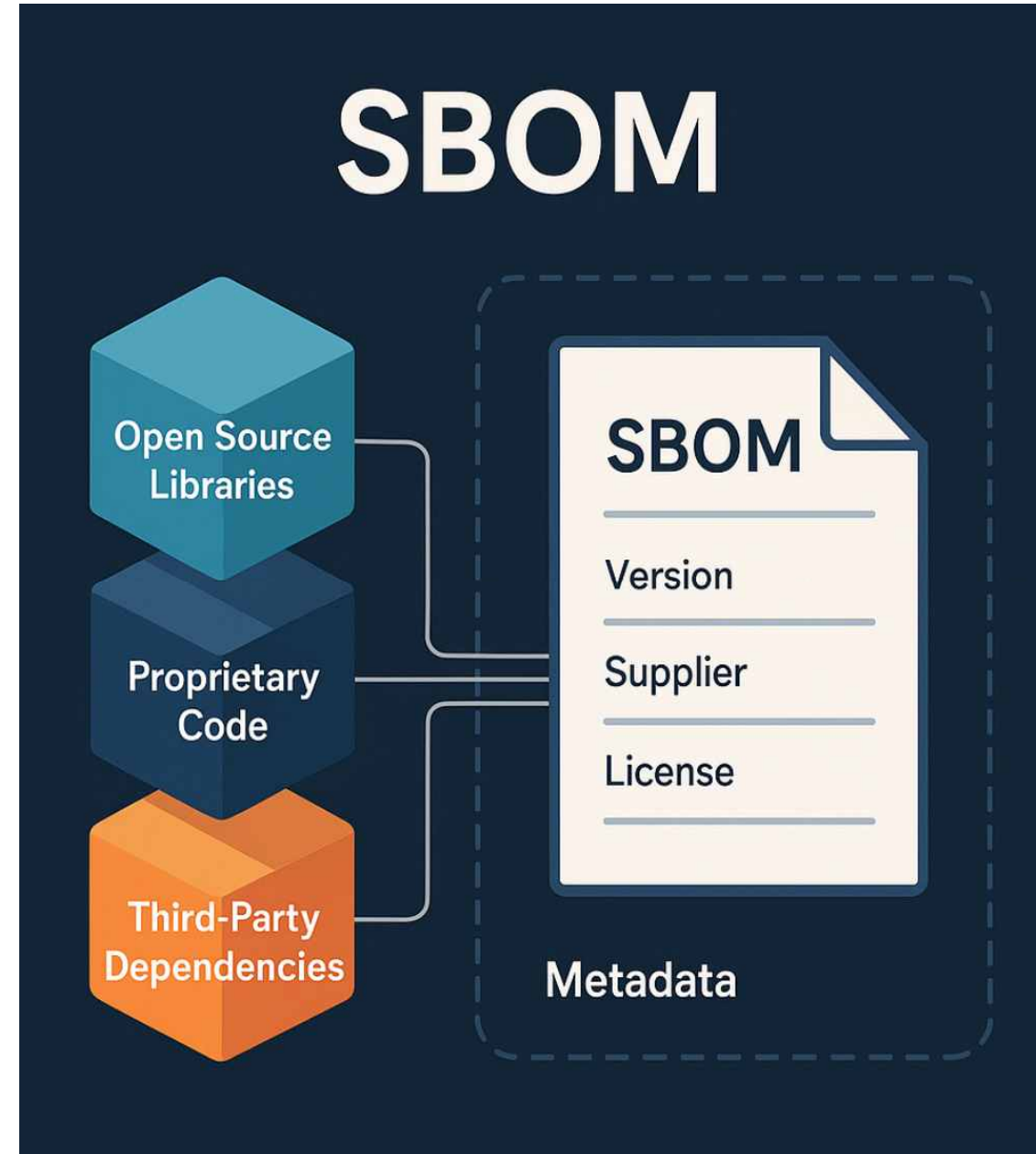
공급망 보안을 위한 SBOM 생성 및 VEX 검증기술 연구

SW재난연구센터 워크숍

2025.07.28.(월) 14:00-15:00

고려대학교 컴퓨터학과 교수 이희조

(heejo@korea.ac.kr, <https://ccs.korea.ac.kr>)





Our Journey Today

1. 문제의식: 왜 SBOM 연구를 시작했는가?
2. 기술적 진화: 오픈소스 분석에서 SBOM 도구까지
3. 국내외 실증: 오픈 플랫폼 구현 및 현장 적용
4. 향후 연구 계획



Prof. Heejo Lee

Korea University

Department of Computer Science and Engineering

heejo@korea.ac.kr



고려대 이희조 교수는

오픈소스 보안 및 SW 취약점 분석 연구의 전문가입니다.

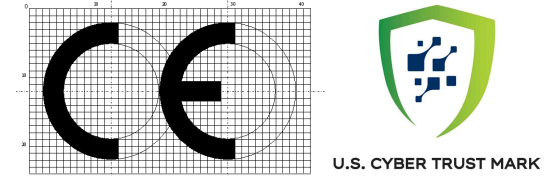
Key Achievements

- 33년 전, 현존하는 국내 최초 화이트 해커 동아리 포스텍 PLUS 창립
- 고려대 소프트웨어보안연구소 (CSSA) 설립 및 국제공동연구 추진 (2015~)
- 보안취약점 자동분석 플랫폼 **IoTcube.net** 런칭 (2016~)
- SCA 솔루션 기업 **래브라도랩스** 공동대표 (2018~)
- 사이버 보안 기업 **안랩** CTO 역임 (2001~2003)
- 대한민국 사이버 보안 정책 자문 활동 (필리핀, 우즈베키스탄 등, 2006~)
- (ISC)² ISLA Award 아시아태평양 Community Service Star 수상 (2016)

1. 문제의식: 왜 SBOM 연구를 시작했는가?

- 보이지 않는 공급망 위협 + 국제 정책 변화가 만든 필연적 대응
 - SolarWinds, Log4j 사태로 “공급망이 공격 경로”라는 현실 확인
 - SBOM (Software Bill of Materials) 통한 투명성과 보안관리 강화
 - 미국 EO, 유럽 CRA, 한국 디지털의료 보안지침 등 규제 본격화
 - 국내외 규제 공통 요구사항: SDL(개발)+SBOM(유통)+VDP(운영)

유럽 CE Mark 및 미국 Cyber Trust Mark



「디지털의료기기 전자적 침해행위 보안지침」(식약처, 2025.04.)

구분	핵심 정책 및 시행 일정	시사점
미국 (EO14028)	<ul style="list-style-type: none"> • US EO 14028 (2021.05), EO 14144 (2025.01) • FDA 의료기기 사이버보안 강화 시행 (2023.10) • Cyber Trust Mark 인증제 시행 (2024~) 	<ul style="list-style-type: none"> • 조달·의료·IoT 등 산업별 대응 필요 • 중소 민간기업의 사전준비 중요 (수출 대응 목적)
유럽 (CRA, NIS2)	<ul style="list-style-type: none"> • 유럽연합 사이버보안 공통규제 NIS2 Directive 보안 강화 2023.01 발효 • CRA (Cyber Resilience Act) 2024.12 발효 → CE Mark 인증 2027 적용 	<ul style="list-style-type: none"> • 제품 인증 = SBOM 포함 • CE 인증 대응 위한 기술 정비 필수
한국	<ul style="list-style-type: none"> • 공급망 보안 가이드라인 1.0 발표, 2024.05 • 디지털의료제품법 시행, 2025.01 	<ul style="list-style-type: none"> • 다양한 분야 법제화 진행 중 • 제도·표준 준비의 골든타임

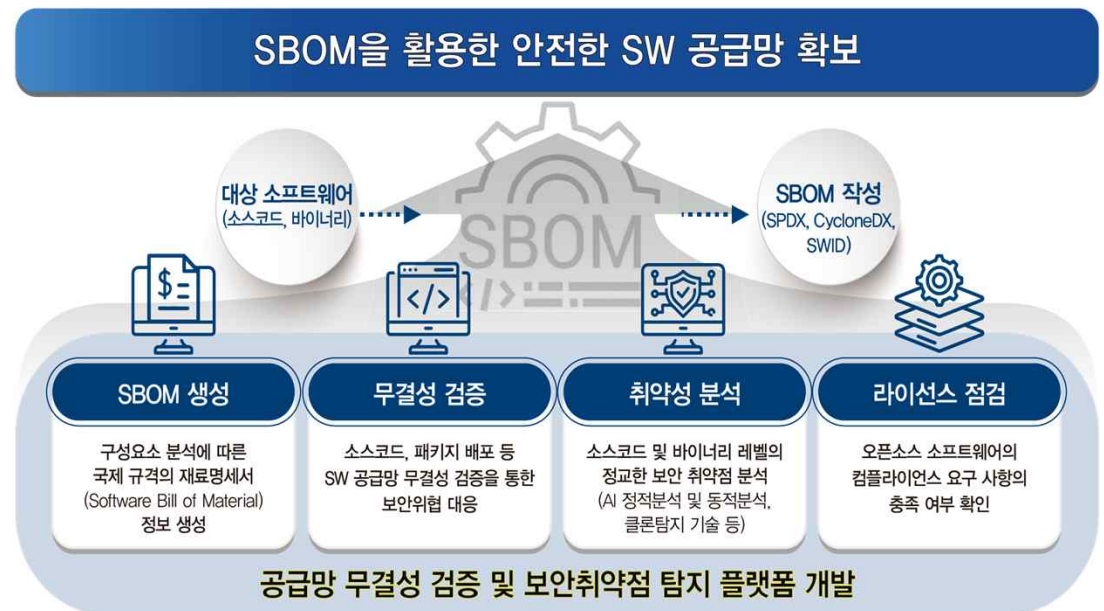
제16조(소프트웨어 구성요소 명세서 관리 활동) ① 디지털의료기기제조
업자들은 디지털의료기기 내 취약점 발견, 보안 및 침해사고 및 이를
해결하기 위한 활동을 수행하는 데 소프트웨어 구성요소 명세서를 활
용할 수 있다.

② 의료서비스제공자는 디지털의료기기에 대해 디지털의료기기제조
업자들이 작성한 소프트웨어 구성요소 명세서를 구매 및 설치 이전에
확인하는 것을 고려할 수 있다.

③ 소프트웨어 구성요소 명세서 정보는 보호되어야 하며 소프트웨어
구성 요소 명세서의 생성, 저장, 송수신 등의 과정에서 데이터 보안을
고려할 수 있다.

1. 문제의식: 왜 SBOM 연구를 시작했는가?

- (과제명) SW공급망 보안을 위한 SBOM 자동생성 및 무결성 검증기술 개발
 - (연구개발기관) 고려대, 강원대, KAIST, 한국정보보호산업협회(KISIA), 2022~2025년
- (목표) SBOM을 활용한 공급망 무결성 검증 및 보안취약점 탐지 플랫폼 개발
 - SBOM 생성 기술
 - 소스코드 기반 취약점 탐지 기술
 - 바이너리 기반 취약점 탐지 기술
 - DB 구축 및 SBOM 실증
- (기대 효과) 국내 최초 SBOM 기술 연구
 - SBOM 기반 취약점 분석 등 핵심 기술 연구
 - 누구나 활용 가능한 오픈 플랫폼 도구 공개
 - 국내외 실증을 통한 산업 현장 적용 및 확산



2. 기술적 진화: 오픈소스 분석에서 SBOM 도구까지

- 고려대 소프트웨어보안연구소 (CSSA, Center for Software Security & Assurance)
 - (비전) 누구나 보안 취약점을 스스로 분석하고 대응할 수 있는 세상을 목표로 2015년 설립
 - 10년간 국제공동연구 플랫폼과 인재양성 경험을 바탕으로 글로벌 연구 거점 성장 (ETH Zurich, CMU, 조지아텍 등)
- 보안취약점 자동분석 플랫폼 IoTcube (아이오티큐브)
 - 2016년 4월 런칭, 141개국 2.6만 명 활용 <https://iotcube.net>
 - 2023년 8월, SBOM 생성 도구 HatBOM 런칭 → 2천 건 활용

중앙일보 2015년 11월 19일 목요일 제2158호

“사이버 보안 인력 태부족... IT-현장 이해하는 ‘융합보안’ 인재 키워야”

융합보안 인재 양성하는 이화조 고려대 융합보안대학원 책임교수

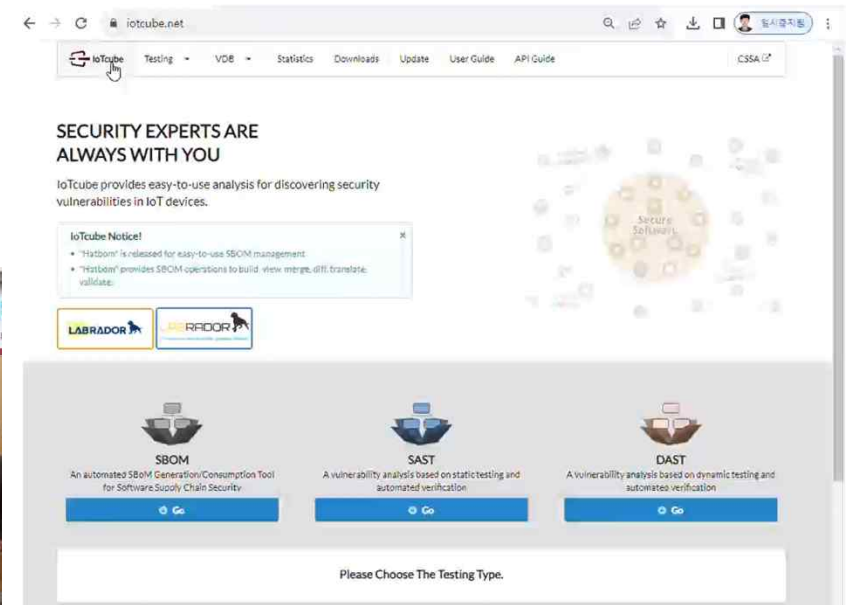
“정보를 활용하기 위해 사이버 보안에 관한 전문 인력(보안 전문가)을 많이 필요로 합니다. 이는 학제 융합의 시대를 맞이한 우리 사회의 현실입니다. IT-현장에 대한 이해가 부족한 보안 전문가의 수요가 증가하고 있습니다.”

이화조 교수는 융합보안대학원 책임교수로서, “융합보안 인재 양성을 위한 융합보안대학원 설립을 위한 12개 연구실 참여에 적극적인 지원과 기업 인턴십-채용 기회 제공을 기쁘게 생각합니다.”

“융합보안 인재 양성을 위한 융합보안대학원 설립을 위한 12개 연구실 참여에 적극적인 지원과 기업 인턴십-채용 기회 제공을 기쁘게 생각합니다.”



“융합보안 인재 양성을 위한 융합보안대학원 설립을 위한 12개 연구실 참여에 적극적인 지원과 기업 인턴십-채용 기회 제공을 기쁘게 생각합니다.”

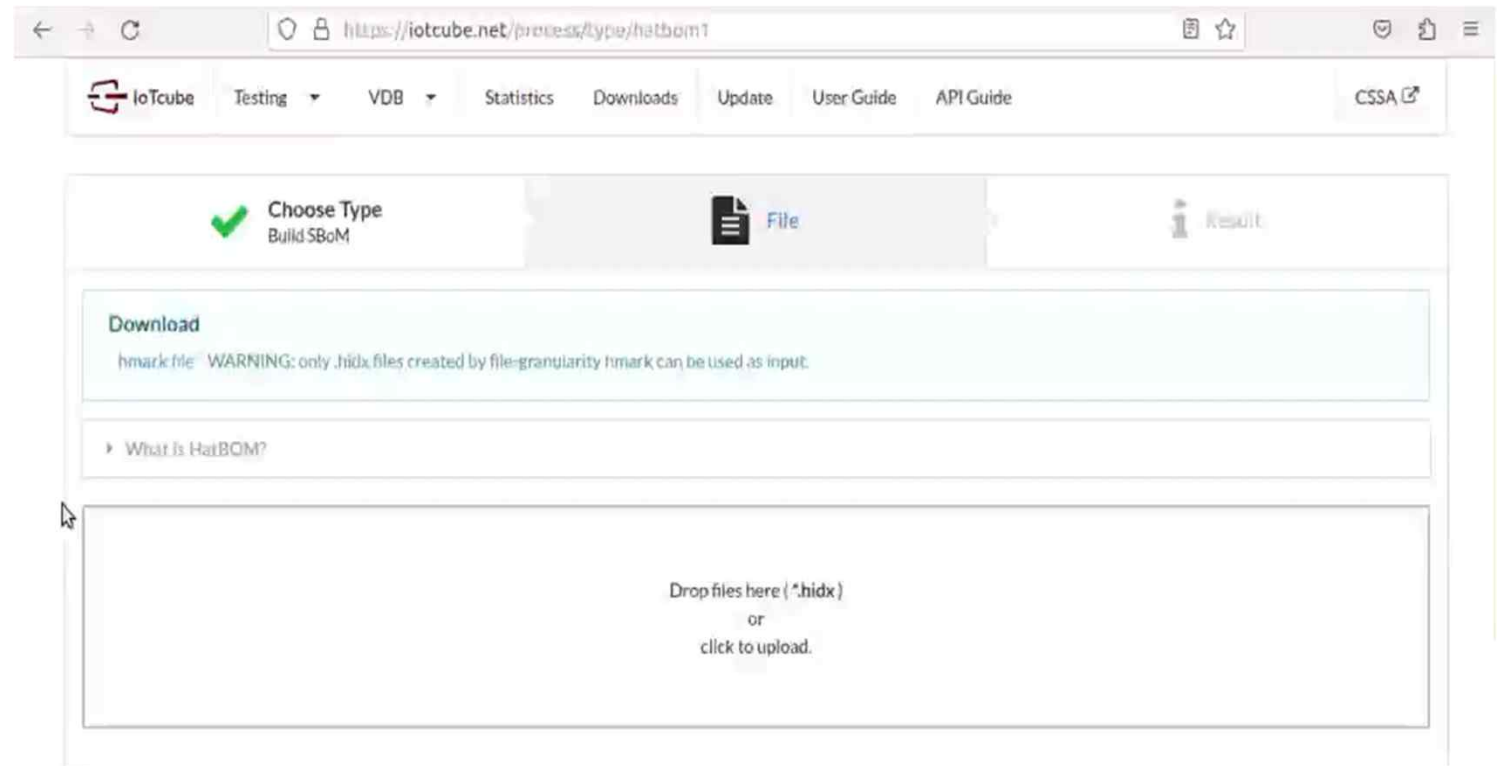
“융합보안 인재 양성을 위한 융합보안대학원 설립을 위한 12개 연구실 참여에 적극적인 지원과 기업 인턴십-채용 기회 제공을 기쁘게 생각합니다.”



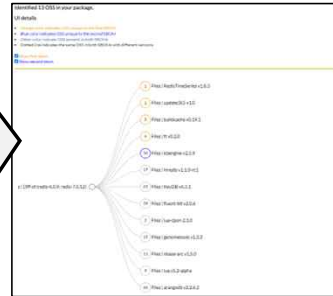
2. 기술적 진화: 오픈소스 분석에서 SBOM 도구까지

“SBOM 개념은 알겠지만... 직접 만들어볼 수는 없을까?”

- 누구나 손쉽게
활용 가능한
SBOM 자동 생성 도구
 - 연구자, 개인용:  IoTcube
IoTcube HatBOM
<https://iotcube.net/hatbom>
 - 기업용:  LABRADOR LABS
래브라도랩스
<https://labradorlabs.ai/>



2. 기술적 진화: 오픈소스 분석에서 SBOM 도구까지



- SBOM 생성, 변환, 병합, 검증이 가능한 4개 도구로 구성
- NTIA에서 정의한 9개 SBOM Tool 연산 항목 중 7개를 충족
- 연산 결과 시각화를 통해 수행 결과를 한눈에 확인 가능

C: CycloneDX SBOM / S: SPDX SBOM / Δ: To be supported

SBOM Tool	F1 (Build)	F2 (Analyze)	F3 (Edit)	F4 (View)	F5 (Diff)	F6 (Import)	F7 (Translate)	F8 (Merge)	F9 (Tool Support)
bomber (DFKM)		C, S		C, S	NOT SUPPORTED				
MS SBOM Tool	S								
Syft	C, S	C, S	C, S						
Tern	C, S								
Aqua Trivy	C, S	C, S	C, S						
CycloneDX CLI			C		C		C, S	C	
HatBOM	C, S	C, S		C, S	C	C, S	C, S	C	Δ

* A part of this table entries were taken from <https://github.com/awesomeSBOM/awesome-sbom>.



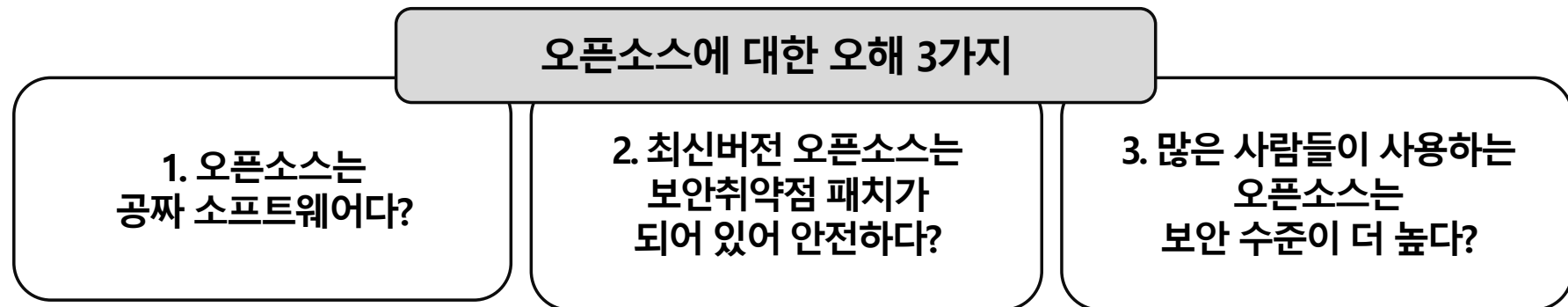
2. 기술적 진화: 오픈소스 분석에서 SBOM 도구까지

- 오픈소스 정의

- 소스코드가 공개되어 라이선스 규칙에 따라 누구나 자유롭게 복사, 수정, 사용, 재배포 할 수 있는 소프트웨어
- 예시 : Linux, TensorFlow, OpenCV, Bitcoin, ROS, AGL 등 (GitHub 5억개 이상 프로젝트 호스팅 중)

- 오픈소스는 기술 혁신의 원동력: AI 프로젝트 증가와 함께 중요성 대거 부각

- "Most major software packages include open source software – including software used by the national security community. Open source software brings unique value, and has unique security challenges, because of its breadth of use and the number of volunteers responsible for its ongoing security maintenance."
(2022년 1월, 미 백악관 소프트웨어 시큐리티 서밋 브리핑 성명)

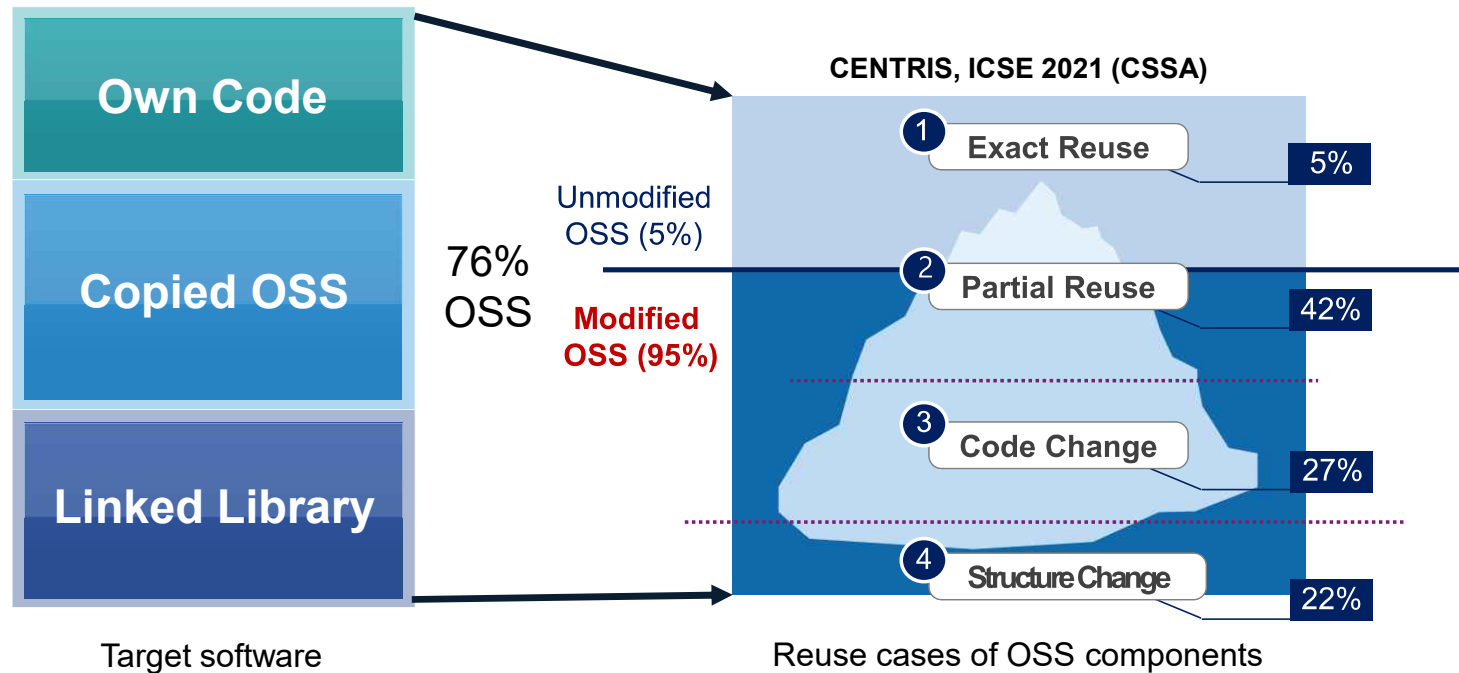




2. 기술적 진화: 오픈소스 분석에서 SBOM 도구까지

• Vulnerability Propagation via Software Reuse

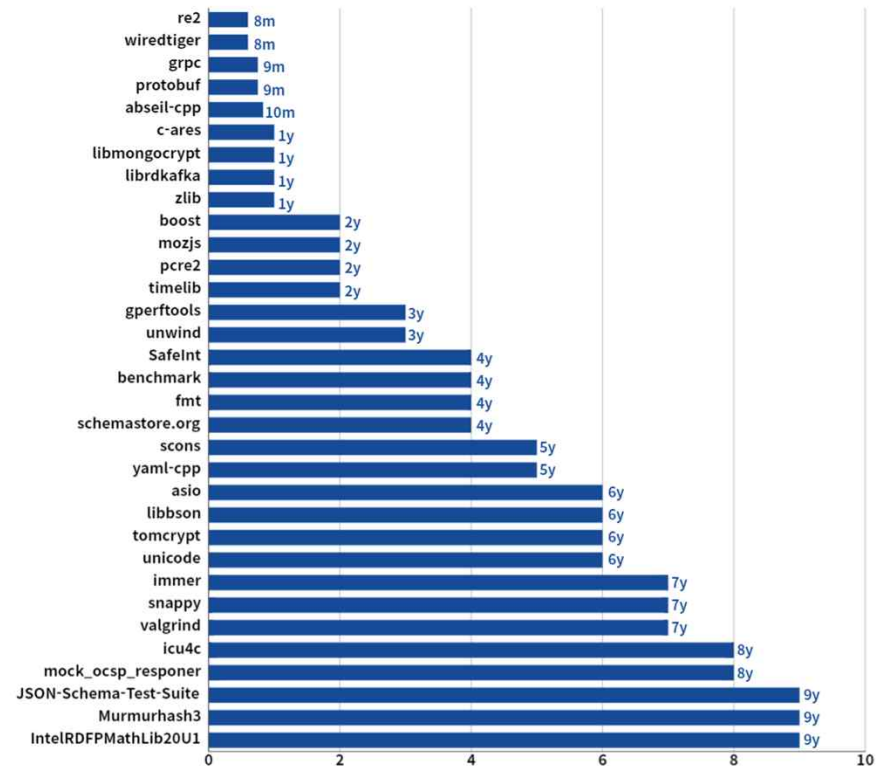
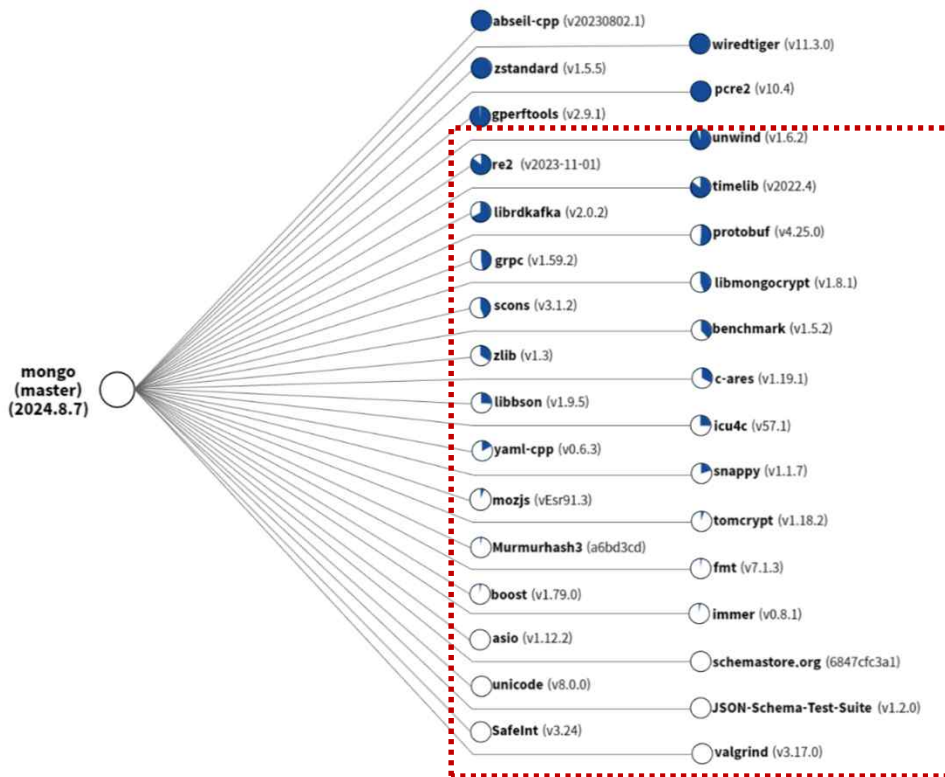
- 76% of codebases are OSS, with 95% being modified or partially used in C/C++ projects (CENTRIS, ICSE 2021)



2. 기술적 진화: 오픈소스 분석에서 SBOM 도구까지

“Are the latest OSS versions always secure?” ❌

- MongoDB contained 36 OSS components, with some unpatched for 9+ years



2. 기술적 진화: 오픈소스 분석에서 SBOM 도구까지

“How do I accurately detect OSS components?”

❖ CENTRIS (ICSE 2021)

- Identifies modified OSS components with 90%+ accuracy

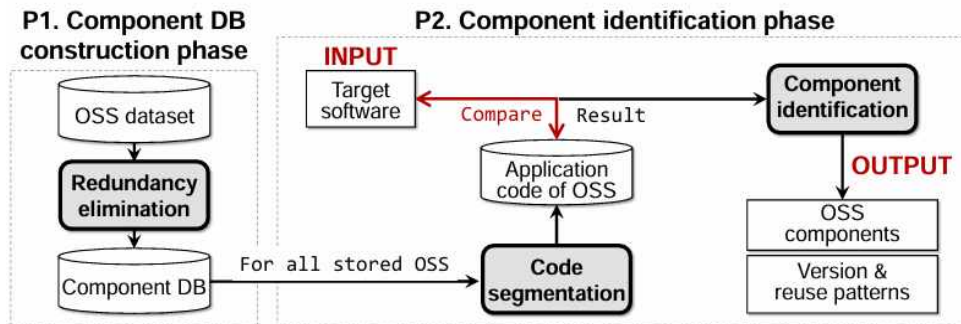


Fig. 2: High-level overview of the workflow of CENTRIS.

※ CENTRIS: A Precise and Scalable Approach for Identifying Modified Open-Source Software Reuse, ICSE 2021

Challenge	Identifying modified OSS is difficult
Solution	Instead of generating signatures from all functions, do redundancy elimination and code segmentation
Impact	Over 90% accuracy (vs. 10% of previous approaches)



2. 기술적 진화: 오픈소스 분석에서 SBOM 도구까지

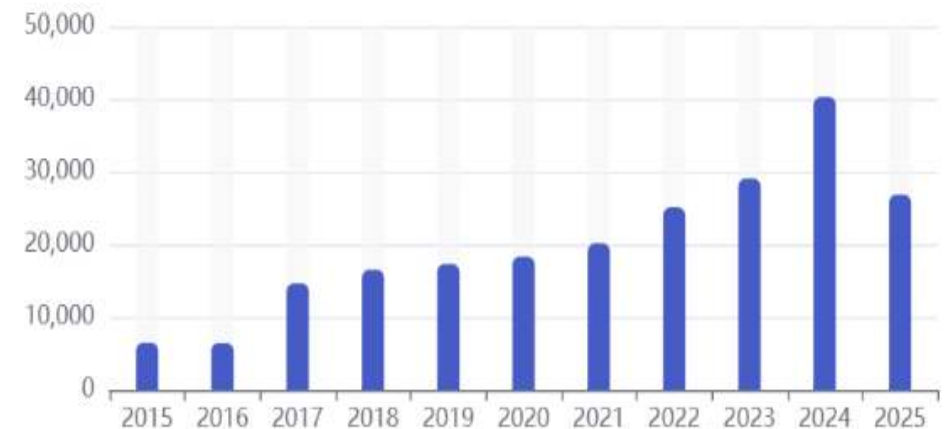
• 보안 취약점이란?

- 예상 못한 입력에 오동작하여 공격에 이용될 수 있는 보안상 약점
- Intel AMT Bug (CVE-2017-5689) Silent Bob is Silent
 - `strcmp(s_pw, e_pw, strlen(e_pw))==0?`

• CVE 취약점이란?

- Common vulnerability and exposure (CVE)는 국제 공통 취약점 식별 체계
- 39개국의 460개 CNA (CVE Numbering Authorities)가 취약점 번호 부여 (2025년 7월 현재 28만 개 이상)
- 미국 MITRE, CISA가 최상위 루트, 국내엔 KISA, 금보원, NAVER, 삼성, LG 등 7개 기관 참여





Number of CVEs by year



※출처: <https://www.cvedetails.com/browse-by-date.php>

2. 기술적 진화: 오픈소스 분석에서 SBOM 도구까지

- 전 세계 상용 소프트웨어의 97%는 오픈소스 사용 중 (Synopsys "OPEN SOURCE SECURITY AND RISK ANALYSIS REPORT", 2025)
 - 오픈소스 등의 공통 취약점을 공격하거나 코드에 접근하여 악성코드 심는 등, 광범위하고 지속적 피해 발생

	구분	일시	CVE 종류	내용
	OpenSSL HeartBleed	2014년	CVE-2014-0160	OpenSSL을 사용하는 웹사이트와 VPN 제품에 취약점으로 원격에서 메모리접근 가능, 캐나다 정부 사이트 공격으로 섰다운 사건 등 발생
	Linux DirtyCOW	2016년	CVE-2016-5195	리눅스 기반 시스템의 루트 권한 획득, 수년간 리눅스 서버 및 IoT 기기 등 공격에 익스플로잇 체인으로 활용
	Log4j 취약점	2021년	CVE-2021-44228	Java 로깅 프레임워크의 취약점으로 다수의 웹사이트 공격이 발생하였고, 어느 제품, 어느 서버가 사용 중인지 파악이 어려워 점검 및 업데이트에 큰 비용 유발
	XZ utils 백도어	2024년	CVE-2024-3094	리눅스 배포판 대부분에 포함된 압축 명령어 "xz"에 백도어를 설치하여 배포, 전세계 수 억대 리눅스 기반 시스템을 "ssh" 명령어로 루트 접근 가능

- CVE-2024-6387 regreSSHion(리그레션) 취약점, 2024년
 - Glibc 기반 리눅스 시스템들의 오픈 SSH 서버 위협 발생, 취약점 익스플로잇 성공 시 공격자는 시스템 전체 장악 가능
 - 2006년 발견되어 패치된 CVE-2006-5051 취약점과 유사 사례이기 때문에, 회귀라는 뜻으로 명칭

이미 완료되어 수정된 오픈소스 취약점도 다시 회귀하여 문제를 일으킬 수 있음 (recurring vulnerabilities)

2. 기술적 진화: 오픈소스 분석에서 SBOM 도구까지

“How do I find hidden vulnerabilities in OSS?”

❖ VUDDY (IEEE S&P 2017)

- Processing units: tokens, blocks, **functions**, files, graphs and others?
- Function hashing finds vulnerabilities 1,000x faster than existing methods

```

Level 1: Formal parameter abstraction.
1 void avg (float FPARAM[], int FPARAM) {
2   static float sum = 0;
3   unsigned int i;
4   for (i = 0; i < FPARAM; i++)
5     sum += FPARAM[i];
6   printf("%f %d", sum/FPARAM, validate(sum));
7 }

Level 2: Local variable name abstraction.
1 void avg (float FPARAM[], int FPARAM) {
2   static float LVAR = 0;
3   unsigned int LVAR;
4   for (LVAR = 0; LVAR < FPARAM; LVAR++)
5     LVAR += FPARAM[LVAR];
6   printf("%f %d", LVAR/FPARAM, validate(LVAR));
7 }

Level 3: Data type abstraction.
1 void avg (float FPARAM[], int FPARAM) {
2   DTYPE LVAR = 0;
3   unsigned DTYPE LVAR;
4   for (LVAR = 0; LVAR < FPARAM; LVAR++)
5     LVAR += FPARAM[LVAR];
6   printf("%f %d", LVAR/FPARAM, validate(LVAR));
7 }

Level 4: Function call abstraction.
1 void avg (float FPARAM[], int FPARAM) {
2   DTYPE LVAR = 0;
3   unsigned DTYPE LVAR;
4   for (LVAR = 0; LVAR < FPARAM; LVAR)
5     LVAR += FPARAM[LVAR];
6   FUNCCALL("%f %d", LVAR/FPARAM, FUNCCALL(LVAR));
7 }

```

Fig. 2: Level-by-level application of abstraction schemes on a sample function.

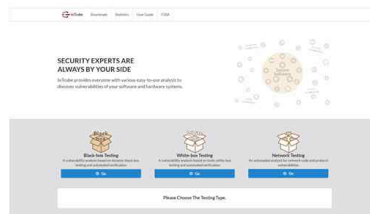


Fig. 10: The main page of IoTcube. The implementation of VUDDY is under the White-box Testing menu.

Rank	CVE	#	Rank	CWE	#
1	CVE-2009-0029	1,377	1	CWE-264	9,305
2	CVE-2016-1575	703	2	CWE-119	8,850
3	CVE-2010-2939	700	3	CWE-399	7,134
4	CVE-2015-2695	676	4	CWE-020	6,751
5	CVE-2006-2313	642	5	CWE-018	2,602
6	CVE-2015-3194	634	6	CWE-362	2,459
7	CVE-2015-5157	618	7	CWE-189	2,454
8	CVE-2016-5424	534	8	CWE-200	2,064
9	CVE-2015-7973	531	9	CWE-310	2,052
10	CVE-2014-3534	503	10	CWE-017	636

Fig. 11: Statistical knowledge obtained by web service for 11 months. Tables show the most frequently detected CVEs and CWEs, respectively. This information is also open to the users.

Challenge	Previous algorithms are slow and inaccurate
Solution	Convert one function into one hash after abstraction and normalization, then compare with the hashes of vuln functions
Impact	1,000x faster with high accuracy

2. 기술적 진화: 오픈소스 분석에서 SBOM 도구까지

“Can I detect vulnerabilities in modified OSS?”

❖ MOVERY (USENIX Security 2022)

- Detects vulnerabilities even in changed codebases

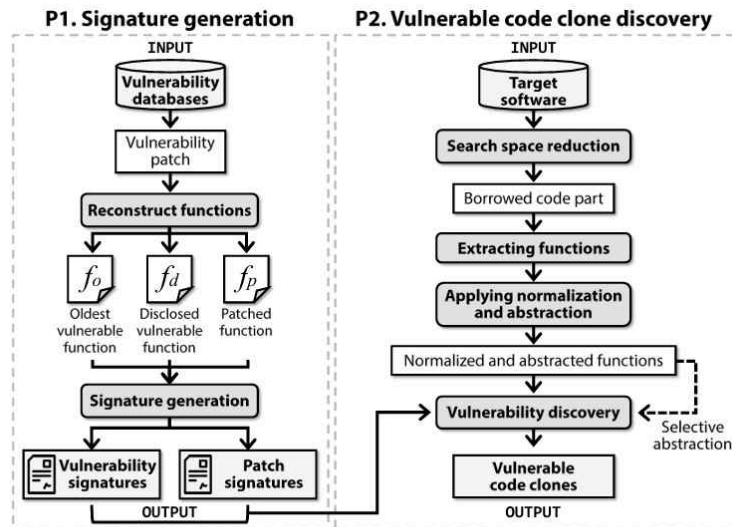


Figure 2: High-level overview of the workflow of MOVERY.

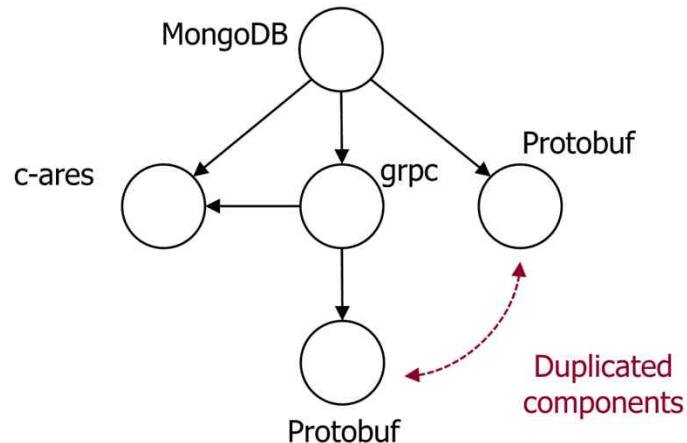
Challenge	Hard to detect vulnerabilities in modified OSS
Solution	Identifies vulnerabilities in changed codebases by making signatures along with vulnerable lines and patched lines in a function
Impact	Higher accuracy than CVE scanning

2. 기술적 진화: 오픈소스 분석에서 SBOM 도구까지

“How do I accurately detect OSS dependencies?”

❖ CNEPS (ICSE 2024)

- Maps complex OSS dependency relationships for accurate risk assessment



Challenge	OSS dependencies are complex
Solution	Maps relationships among components for risk assessment and update management
Impact	Improves SBOM accuracy with 75% more dependency discovery



3. 국내외 실증: 오픈 플랫폼 구현 및 현장 적용

- IoTcube HatBOM 도구를 활용하여, 국내외 실증 진행

(국내) KISIA 협력, 3개 정보보호 기업 실증 진행

<https://www.kisia.or.kr/announcement/association/706/>

- HatBOM으로 탐지된 자사 솔루션 취약점을 각 기업 개발자와 교차 검증
- 기존 OSS 인벤토리 관리 방식에서 → SBOM 기반 체계적 관리로 내부 전환 유도

(해외) 美 카네기멜런대 SEI 주관 SBOM Harmonization Plugfest 2024 참가

- 美 CISA&CMU, SBOM 상호 호환성 글로벌 점검
- 고려대, HatBOM으로 생성한 SBOM 2건 제출 → 240+개 SBOM 분석 보고서 발표 예정 (2025.07)

SBOM Harmonization Plugfest 2024

NOVEMBER 19 - DECEMBER 15, 2024 | VIRTUAL



구분	실증결과에 대한 평가
총평 및 시사점	<ul style="list-style-type: none"> - R&D 연구개발된 SBOM 생성도구를 통해 솔루션의 SBOM을 생성하여 그 도구와 명세서가 NTIA 기준에 맞게 정상적으로 작동하고 대부분의 컴포넌트 및 취약점에 대한 성공적 검출을 확인 - 일부 미탐 라이브러리 및 컴포넌트, 취약점에 대해서는 기능향상을 위한 기술적 지원 필요 - 공통적으로 SBOM에 대한 가이드 및 지침 필요, 지원언어 확대 등을 개선방안으로 제안 - SBOM의 신뢰성 향상을 위한 표준 준수, 담당기관 지정, 중소기업 교육 등 SBOM 신뢰성 확보와 역량강화를 위한 정책 지원 필요
Lesson Learned	<ul style="list-style-type: none"> - 미국, 유럽 등 SBOM 컴플라이언스 시행에 따른 국제 SW 무역규제 대비에 막대한 불안감이 있었으나 이번 실증사업 참가를 통해 자체 내부 검토 필요성을 인식하는 계기가 됨 - 기업에서 보유한 SW OSS 구성요소에 대해 투명하게 관리할 수 있는 관리체계(담당부서)의 필요성 인식 - SW공급망 공격에 대한 대비를 위해 사용중인 OSS라이브러리와 구성요소를 목록화하고 취약점을 체계적으로 관리해야하는 필요성을 인식 - 식별된 OSS의 보안 취약점을 사전에 탐지하여 예방할 수 있는 조치를 검토하게 된 계기가 됨 - 실증사업 참여를 통해 자체적으로 사용중인 OSS를 목록화하고 업데이트를 수행하는 계기가 됨 - SBOM 도구를 통한 OSS 확인과 취약점 개선을 통한 안정적 SW경쟁력 확보방안의 가능성을 확인



3. 국내외 실증: 오픈 플랫폼 구현 및 현장 적용

- SBOM 솔루션 시장 적용 사례 

항목	구축 사례	솔루션 예시
CI/CD 환경 적용	<ul style="list-style-type: none">• 개발 전 오픈소스 허용 목록 관리• CI/CD 전주기에서 취약점 자동 점검 및 SBOM 연계 관리	SCA
오픈소스 거버넌스 구축	<ul style="list-style-type: none">• 금융권 등에서 오픈소스 반입 시스템에 SBOM 활용• 전사 조직 및 결제 시스템과 연계된 취약점 관리 체계 확립	IVAS
SBOM 공급망 체계 구축	<ul style="list-style-type: none">• 협력사-제조사-고객사 SBOM 교환 플랫폼 구축	SCM
서버 오픈소스 취약점 관리	<ul style="list-style-type: none">• 서버 취약점 실시간 모니터링 및 패치 관리• AI 모델 취약점 분석	Server Care

※ 자료 제공: 래브라도랩스 솔루션 활용 현황 <https://labradorlabs.ai/>



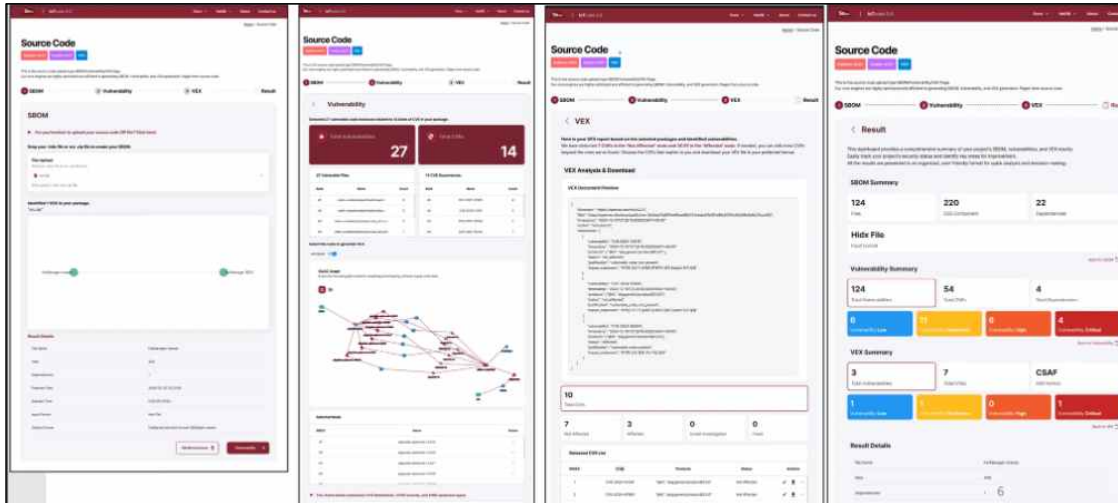
4. 향후 연구 계획

- **SBOM을 넘어, VEX까지 확장된 국제공동연구**
 - 2024년부터 한-미-스위스, 3개국 국제공동연구 진행 (2024.07.~2026.12.)
 - 고려대, 카이스트, Georgia Tech, ETH, Northeastern Univ. 및 OpenSSF 참여
 - SBOM + VEX 기반 위협 대응 기술 문서 자동화 연구
→ SBOM 유지 비용 절감, 대응 효율성 제고
- **2025년 8월, IoTcube(아이오티큐브) 2.0 공개**
 - 2023년 8월, 공개형 SBOM 자동 생성 관리 도구 HatBOM 런칭
 - 2025년 8월 26일(화), 컨퍼런스 통해 IoTcube HatBOM 2.0 플랫폼 공개 예정



4. 향후 연구 계획

- 제9회 IoTcube 컨퍼런스 (8월 26일 화, 동대문 JW메리어트)
 - 美 노스이스턴대 Archimedes Center와 의료기기 보안 주제로 강연 및 비공개 SBOM 운영 사례 세미나 개최
- Next Platform 'HatBOM'
 - One-Step Process (SBOM→Vulnerability→VEX) 분석





**Archimedes Center for
Health Care and Medical Device Cybersecurity
at Northeastern University**



KOREA UNIVERSITY
Center for Software Security and Assurance

**ARCHIMEDES INTERNATIONAL HEALTHCARE
SECURITY WEEK**

IoTcube Conference 2025

Tuesday, August 26, 2025 | 10:00AM - 6:00 PM (KST)
Grand Ballroom, B1 Level, JW Marriott Dongdaemun Square
Seoul, South Korea
(In-person only)

SPEAKERS:


KEVIN FU
Professor
Northeastern University


HEEJO LEE
Professor
Korea University


CHRISTIAN DAMEFF
Emergency Physician
UC San Diego


WENYUAN XU
Professor
Zhejiang University


SANG KIL CHA
Professor
KAIST


JEFF TULLY
Anesthesiologist
UC San Diego


JACK KUFAHL
CISO
Michigan Medicine


YUSEUNG KIM
Head of System S/W
Group & Corp. VP
Samsung Electronics

HOSTS:

CSSA (Center for Software Security and Assurance), Korea University (South Korea), and
Archimedes Center for Healthcare and Medical Device Cybersecurity, Northeastern University (USA)

**SCAN TO
LEARN MORE:**



https://cssa.korea.edu/cssa_en/conference/intro.do

jihyeonlee@korea.ac.kr or archimedes@northeastern.edu

secure-medicine.org



"SBOM에서 시작된 여정이 이제는 VEX,
그리고 글로벌 보안 대응 생태계로 확장되고 있습니다.
안전한 공급망 보안을 위해 더 많은 협력과 참여를 기대합니다."

Q&A



Email: Prof. Heejo Lee, heejo@korea.ac.kr
CSSA Center, cssa@korea.ac.kr



Website: <https://cssa.korea.ac.kr>