전기차 충전 인프라 오픈소스 소프트웨어에서의 취약한 코드 복제 탐지: VUDDY 를 활용한 사례 연구

전세옥 [1], 장현지 [2], 정민수 [3], 이서준 [2], 오다영 [2], 최광훈 [4], 김석휘 [5], 조성우 [6], 김정미 [5]
[1] 전남대학교 정보보안융합학과 석사과정
[2] 전남대학교 인공지능학부
[3] 전남대학교 소프트웨어공학과
[4] 전남대학교 정보보안융합학과 교수
[5] (주)케이사인
[6] (주)에이아이딥

sojeon@jnu.ac.kr, gka1225@jnu.ac.kr, jeungminsooo@gmail.com, tjwns1300@jnu.ac.kr, dalucky3072@gmail.com, kwanghoonchoi@jnu.ac.kr, sukwhi809@ksign.com, swcho@aideep.ai, jenny@ksign.com

Detecting Vulnerable OSS Code Clones in Electric Vehicle Charging Infrastructure Using VUDDY

Se-Ok Jeon[1], Hyeonji Jang[2], Min-Soo Jeung[3], SeoJun Lee[2], Da-Young Oh[2], Kwang-hoon Choi[4], Seokhwi Kim[5], Sungwoo Cho[6], Jwongmi Kim[5]
[1]Dept. of Information Security Convergence, Chonnam National University
[2]Dept. of Artificial Intelligence, Chonnam National University
[3]Dept. of Software Engineering, Chonnam National University
[4] Dept. of Information Security Convergence, Chonnam National University
[5] KSIGN
[6] AiDeep

요 약

오픈소스 코드 재사용은 비용 절감과 생산성을 높이지만, 관리의 복잡성으로 인해 취약점이 발생할 수 있다. 본 논문은 전기차 충전 인프라에서 사용되는 오픈소스 소프트웨어의 취약한 코드 복제를 탐지하는 방법을 제안한다. VUDDY 와 IoTCube 를 활용해 분석한 결과, open-ocpp 에서 zlib 복제로 인한 취약한 코드 클론 사례가 발견되었으며, 이는 CVE-2016-9840 및 CVE-2016-9841 과 관련이 있었다. 연구는 코드 복제 취약점 탐지를 통한 전기차 충전 인프라 보안 강화 가능성을 보여주었다.

1. 서론

오픈소스 코드 재사용은 비용 절감과 생산성 향상에 기여하면서 널리 확산되고 있다. 그러나 의존하는 오픈소스의 양이 증가하면서 관리가 복잡해지는 문제가 발생한다. 실제로 2024 년 오픈소스 보안 및 위험 분석 보고서[1]에 따르면, 상용 코드베이스 1067 개 중 84%에서 오픈소스 취약점이 발견되었다. 리눅스의 Dirty Cow 사례[2]는 코드 재사용으로 인한 관리 문제를 잘 보여준다. 2005 년에 패치된 리눅스 커널의 권한 상승 취약점이 코드 복제 과정에서 우분투와 데비안으로 전파되었고, 2016 년에 다시 발견되었다. 이 취약점은 안드로이드 운영체제까지 확산되어 그 당시 모든 안드로이드 버전이 루팅[3]될 수 있는 심각한 보안 문제를 초래했다.

이와 같은 취약한 코드 복제 문제를 해결하기 위해 다양한 연구가 이루어졌다. CCFinder[4]는 동일한 복제 코드(type 1)를 탐지하는 데 중점을 두었고, VUDDY[5]는 식별자가 변경된 복제(type 2)를 탐지하기 위해 추상화 기법을 도입했다. Movery[6] 는 복제 후 코드가 재구성된 경우(type 3)까지 탐지하는 방법을 제시했다.

전 세계적으로 전기차 시장이 급격히 성장함에 따라 충전 인프라도 빠르게 확산되고 있다. 이로 인해 전기차 충전 인프라는 사이버 보안 위협의 새로운 공격 표적으로 떠오르고 있다. 실제로 Electrify America 의 충전소가 해킹 되어 원격 제어와 액세스 권한을 탈취당한 사례[7]가 있다.

따라서 본 논문에서는 전기차 충전 인프라에서 사용되는 소프트웨어의 보안을 강화하기 위해, 취약한 코드 복제 탐지를 통한 취약점 점검을 제안한다. 이를 통해 전기차 충전 인프라에서 주로 사용되는 오픈소스 프로젝트에서 취약한 코드 복제 사례를 성공적으로 탐지하는 기여를 했다.

논문은 다음과 같이 구성된다. 2 절에서는 관련 연구를 설명하고, 3 절에서는 취약한 코드 복제 탐지 방법의 세부 사항을 다룬다. 4 절에서는 탐지 결과에 대한 평가를, 5 절에서는 그 한계를 논의한다. 마지막으로 6 절에서 결론을 제시한다.

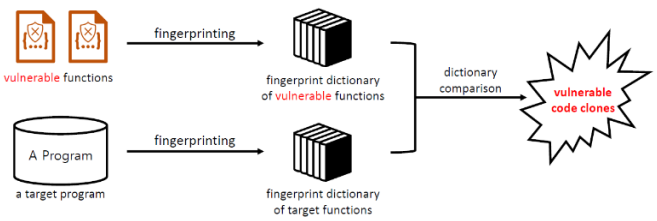
2. 관련 연구

CCFinder 는 토큰 수준의 매칭을 통해 type 1 클론 탐지에서 높은 정확도를 보였지만, 복잡한 소프트웨어에서는 확장성에 한계를 보였다. VUDDY 는 취약점 패치 기반의 type 2 클론을 높은 정확도로 탐지하면서 확장성이 뛰어나다. MOVERY 는 취약점 패치의 히스토리를 고려하며 type 3 클론까지 탐지할 수 있다. 이 논문에서는 IoTCube[1]에 이미 구축되어 있는 취약점 데이터베이스를 활용할 수 있는 VUDDY 를 선택하였다. 다음 절에서 설명하겠지만, VUDDY 는 전기차 충전기 관련 오픈소스 소프트웨어에서도 잘 적용되었다.

Gaetano 등 은 전기차 충전소(EVCS)와 중앙 관리 시스템(CMS) 간 통신에 사용되는 OCPP 오픈소스 구현체를 대상으로 퍼징 테스트를 수행해 DoS 취약점을 발견했다.[8] Ashwin 등 은 RISE-V2G 와 같은 오픈소스를 활용해 전기차와 전기차 공급 장비(EVSE) 간 ISO 15118 통신 시스템을 구축하고 퍼징 테스트를 통해 취약점을 분석했다.[9]

3. 취약한 코드 복제 탐지 방법

취약한 코드 복제 탐지는 함수 추출 및 딕셔너리 생성 그리고 취약점 딕셔너리와의 비교로 구성되어 있다.



(그림 1) 취약한 코드 복제 탐지 방법 개요.

[1] https://iotcube.net

i. 함수 추출 및 딕셔너리 생성

그림 1 왼쪽 아래와 같이 VUDDY 에 실험 대상 소프트웨어의 코드 경로를 입력하면, 파서가 각 파일에서 함수를 수집한다. 수집된 함수는 설정된 옵션에 따라 추상화 및 정규화 과정을 거치며, 함수의 문자열 길이와 해시값이 계산된다. 이때 문자열 길이는 딕셔너리의 키로, 해시값은 값으로 저장되어 함수 핑거프린트 딕셔너리가 생성된다.

ii. 외부 취약점 데이터베이스 (iotcube.net) 활용

그림 1 왼쪽 위에는 취약한 함수들이 있다. 마찬가지로 취약한 함수로부터 데이터베이스를 만들어야하는데, 이번 연구에는 iotcube 가 만들어놓은 취약점 데이터베이스를 활용했다. 취약점 데이터베이스는 커밋 로그에서 CVE 를 검색해 얻은 패치를 기반으로 만들어진다. 패치되기 전의 함수 문자열을 추출해, 해당 문자열의 길이와 해시값을 딕셔너리에 저장하는 방식으로 취약한 함수 데이터베이스가 구성된다.

iii. 딕셔너리와 취약점 데이터베이스 비교

VUDDY 로 생성한 함수 딕셔너리를 iotcube 에 업로드하면, 취약한 함수가 포함되어 있는지 비교를 하고, 그 결과는 웹에서 확인할 수 있다. 다만, 오픈소스 VUDDY 와의 형식 호환 문제로 인해 IoTCube 에서는 최신 버전의 VUDDY 를 사용해야 한다.

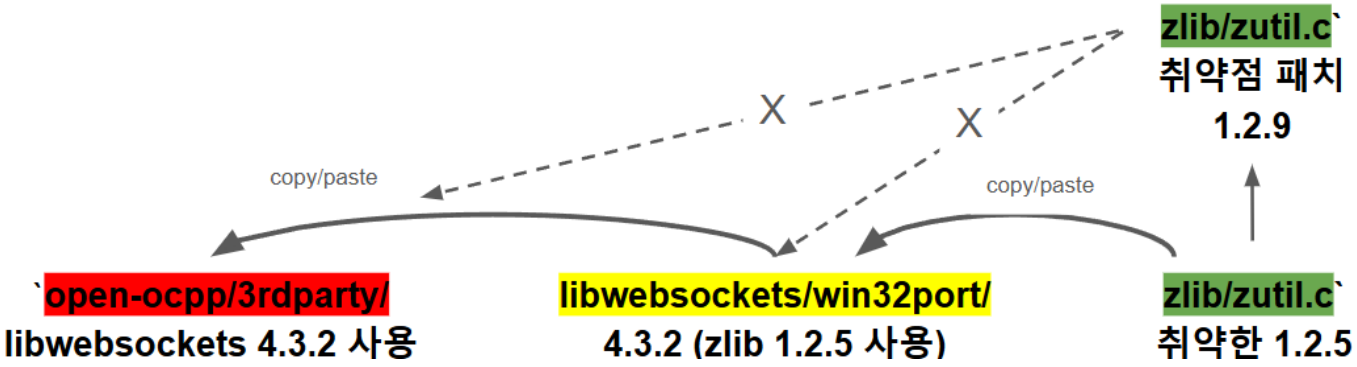
4. 실험 및 평가

본 논문에서 수행한 실험 절차는 다음과 같다.

1. 실험 대상 소프트웨어를 VUDDY 에 입력하여 함수들을 추출하고, 추상화 및 정규화를 통해 각 함수의 길이와 핑거프린트를 생성한다.
2. 생성한 함수 핑거프린트 딕셔너리를 IoTCube 취약점 데이터베이스를 활용하여 입력 소프트웨어에서의 취약한 함수가 사용되었는지 탐지한다.

<표 1> 실험 대상

유형	이름
OCPP	libocpp
	microocpp
	open-ocpp
ISO-15118	openv2g
	ext-openv2g
firmware	open_evse



(그림 2) 결과 분석.

평가는 표 1 과 같이 전기차 충전 인프라와 관련된 총 6 개의 오픈소스 소프트웨어를 대상으로 진행되었다. VUDDY 를 사용하여 각 소프트웨어의 함수 핑거프린트 딕셔너리를 생성하고, iotcube 를 활용해 분석한 결과, open-ocpp 에서 1 건의 취약한 코드 복제 사례가 탐지되었다.

탐지된 취약한 코드 복제는 zutil.c 파일에서 발견되었으며, open-ocpp 가 zlib 코드를 복제해 사용하면서 최신 버전으로 업데이트하지 않아 발생한 문제였다. open-ocpp 는 libwebsockets 프로젝트의 4.3.2 버전을 복제하여 사용했는데, 그 결과 libwebsockets 4.3.2 버전이 취약한 zlib 1.2.5 버전을 포함하게 되면서 취약한 코드가 복제 및 전파된 것이다.

주어진 버전 정보를 기반으로 추가 분석한 결과, open-ocpp 는 CVE-2016-9840 과 CVE-2016-9841 에 해당하는 취약한 함수 또한 복제된 것으로 확인되었다.

5. 한계

IotCube 의 데이터베이스가 비공개 되어 있어, 탐지된 취약점 패치의 출처를 확인할 수 없었다. 이로 인해 탐지 결과를 해석하는 과정에서 어려움이 있었다. 또한 VUDDY 는 C/C++ 코드만 처리할 수 있어, 다양한 언어로 작성된 오픈소스 프로젝트의 경우 취약점을 점검하는데 제한이 있었다.

6. 결론

본 연구에서는 전기차 충전 인프라에서 사용되는 오픈소스 소프트웨어의 취약한 코드 복제를 탐지하여 보안을 강화하는 방법을 제시했다. VUDDY 와 iotcube 를 활용한 분석 결과, open-ocpp 에서 zlib 복제로 인한 취약한 코드 클론이 발견되었으며, 이는 CVE-2016-9840 및 CVE-2016-9841 과 관련된 심각한 보안 취약점을 포함하고 있었다. 이러한 결과는 전기차 충전 인프라와 같은 복잡한 시스템에서 코드 복제에 의한 보안 위험이 존재함을 확인시켜주었다.

향후 연구에서는 VUDDY 의 파서를 개선하여 C/C++ 외에도 다양한 프로그래밍 언어를 지원하도록 확장할 필요가 있다. 이를 통해 더 많은 오픈소스 프로젝트에 대한 취약점 점검을 수행함으로써, 전기차 충전 인프라 뿐만 아니라 다양한 산업 분야에서 오픈소스 소프트웨어의 보안성을 강화할 수 있을 것이다.

Acknowledgement

본 연구에 필요한 IoTCube 를 공개해 주신 CSSA(고려대학교 소프트웨어보안연구소)에 감사드립니다.

본 연구는 과학기술정보통신부 및 정보통신기획평가원의 융합보안핵심인재양성사업의 연구 결과로 수행되었음 “ (IITP-2024-RS-2022-II221203).

본 연구는 한국인터넷진흥원(KISA)-정보보안 특성화대학 지원사업의 지원을 받아 수행된 연구임. 본 연구는 과학기술정보통신부 및 정보통신기획평가원의 인공지능융합혁신인재양성사업 연구 결과로 수행되었음(IITP-2023-RS-2023-00256629).

이 논문은 2024 년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구결과임 (RS-2024-00398993, 전기자동차 충전기 보안위협 대응 기술 개발)

참고문헌

[1] “2024 Open Source Security and Risk Analysis Report (OSSRA) | Synopsys”.

[2] “Linux 커널에서 ‘Dirty Cow(CVE-2016-5195)’ 제로데이 취약점 발견!”, 이스트시큐리티 알약 블로그.

[3] “리눅스 커널 취약점 Dirty Cow(CVE-2016-5195) 통해 모든 버전의 안드로이드 루팅 가능해”, 이스

트시큐리티 알약 블로그.
[4] T. Kamiya, S. Kusumoto and K. Inoue, “CCFinder: a multilinguistic token-based code clone detection system for large scale source code”, *IIEEE Trans. Software Eng.*, vol 28, issue 7, pp 654–670.
[5] S. Kim, S. Woo, H. Lee and H. Oh, “VUDDY: A Scalable Approach for Vulnerable Code Clone Discovery”, in *2017 IEEE Symposium on Security and Privacy (SP)*, 5 2017, pp 595–614.
[6] S. Woo, H. Hong, E. Choi and H. Lee, “MOVERY: A Precise Approach for Modified Vulnerable Code Clone Discovery from Modified Open-Source Software Components”, presented at the 31st USENIX Security Symposium (USENIX Security 22), 2022, pp 3037–3053.
[7] M. Akuchie, “Hacked Electrify America Charger Exposes Major Cybersecurity Risk”, ScreenRant.
[8] G. Coppoletta, A. Kaur, N. Valizadeh, O. Rana, R. Gjomemo and V. N. Venkatakrishnan, “OCPPStorm: A Comprehensive Fuzzing Tool for OCPP Implementations”, Network and Distributed System Security Symposium (NDSS 2024), San Diego, USA, 2024.
[9] A. Nambiar, Z. B. Celik, R. Gerdes and A. Bianchi, “Poster: PLUG&CHECK: Finding Bugs in ISO15118 Implementations with EVFUZZ”, Network and Distributed System Security Symposium (NDSS 2024), San Diego, USA, 2024.