JKSCI

한국컴퓨터정보학회논문지
Journal of The Korea Society of Computer and Information
Vol. 29 No. 11, pp. 173-185, November 2024
https://doi.org/10.9708/jksci.2024.29.11.173

A Study on the Importance of Control Items of NIST SP 800-53 by Mapping CVE and STIG/SRG

Se-Eun Kim*, Hyo-Beom Ahn**

*Student, Division of Artificial Intelligence, Kongju National University, Chonan, Korea
**Professor, Division of Artificial Intelligence, Kongju National University, Chonan, Korea

[Abstract]

The U.S. federal government has established NIST SP 800-53 in response to the need for vulnerability management, and MITRE manages security vulnerabilities through CVE numbers. Although the relationship between NIST SP 800-53 and CVE is a crucial factor in vulnerability management, it is not clearly defined, making it challenging for security managers to identify control items that address the latest vulnerabilities. This study aims to analyze the relationship between NIST SP 800-53 and CVE to establish prioritization for evaluating security control items. Controls that are frequently associated with CVE should be prioritized for evaluation and improvement. The study derived the relevance between NIST SP 800-53 security controls through mapping CVE to STIG/SRG and used SecBERT, CyBERT, and RankT5 models to automate this mapping. The results confirmed the need to prioritize the improvement of specific security controls.

▸ **Key words**: Vulnerability Management, NIST SP 800-53, CVE, Security Control, Automation Mapping

[요 약]

취약성 관리의 필요성에 따라 미국 연방 정부는 NIST SP 800-53을 마련했고, MITRE는 CVE 번호를 통해 보안 취약점을 관리하고 있다. NIST SP 800-53과 CVE 간의 연관성은 취약성 관리에 중요한 요소지만, 명확히 정의되어 있지 않아 보안 관리자들이 최신 취약점에 맞는 통제항목을 파악하기 어렵다. 본 연구는 NIST SP 800-53과 CVE 간의 연관성을 분석하여 보안 통제항목의 평가 우선순위를 설정하는 데 목적이 있다. CVE와 많이 연결된 통제항목을 우선 평가하고 개선해야 한다. 연구는 CVE와 STIG/SRG 간 매핑을 통해 NIST SP 800-53 보안 통제와의 관련성을 도출하였으며, SecBERT, CyBERT, RankT5 모델을 사용해 매핑을 자동화하였다. 결과적으로, 특정 보안 통제를 우선적으로 개선해야 할 필요성을 확인하였다.

▸ **주제어**: 취약점 관리, NIST SP 800-53, CVE, 보안 제어, 자동화 매핑

• First Author: Se-Eun Kim, Corresponding Author: Hyo-Beom Ahn
*Se-Eun Kim (longlngs@naver.com), Division of Artificial Intelligence, Kongju National University
**Hyo-Beom Ahn (hbahn@kongju.ac.kr), Division of Artificial Intelligence, Kongju National University
• Received: 2024. 09. 30, Revised: 2024. 10. 28, Accepted: 2024. 10. 28.

http://www.ksci.re.kr pISSN:1598-849X | eISSN:2383-9945

I. Introduction

사이버 공격의 지속적인 증가로 인해 정보 시스템의 보안 평가와 취약점 관리는 필수적인 요소로 자리 잡고 있다. CVE Details에 따르면, 최근 몇 년간 CVE(Common Vulnerabilities and Exposures) 취약점의 수는 급격히 증가하였으며, 2019년에는 약 8,682건이 보고되었으나, 2023년에는 16,628건으로 91.5% 증가하였다[1]. 이러한 제로데이 취약점의 급증에 대응하기 위해서는 철저한 정보보안 체계를 마련하고 신속한 조치가 필요하다.

미국 연방 정부는 정보시스템 보안 평가를 위해 NIST SP 800-53을 제정하였고, MITRE는 CVE 번호를 통해 공개된 보안 취약점을 식별하고 관리하며, CVE는 각 취약점에 고유한 식별 번호를 부여하여 전 세계적으로 활용되는 체계이다. 그러나 NIST SP 800-53과 CVE는 각각 보안 평가와 취약점 관리의 핵심 요소이나, 두 표준 간의 연관성이 명확히 정의되지 않아, 보안 관리자들이 최신 취약점에 적합한 통제항목을 파악하는 데 어려움을 겪고 있다.

기존 연구들에서는 MITRE ATT&CK와 다른 보안 프레임워크 간의 매핑에 집중해왔으나, NIST SP 800-53과 CVE 간의 직접적인 매핑 연구는 부족한 상황이다.

본 논문에서는 NIST SP 800-53 보안 통제항목과 CVE 간의 연관성을 분석하여 보안 통제항목의 평가 우선순위를 설정하는 것을 목표로 하고, 이 연구를 위해 CVE와 STIG(Security Technical Implementation Guide), SRG(Security Requirements Guide) 간의 관계를 매핑하고, NIST SP 800-53 보안 통제와의 연결성을 도출하여 평가의 우선순위를 설정하기 위한 방법을 제안한다.

이 제안을 위한 연구방법은 세 가지 주요 분석을 포함한다. 첫째로, NIST SP 800-53의 기술적 통제 항목을 CCI(Control Correlation Identifier) 및 SRG/STIG와 매핑하여 연관성을 파악하였다. 둘째로, OVAL(Open Vulnerability and Assessment Language) 파일을 활용해 STIG와 CVE 간의 매핑을 수행하여 각 취약점에 대응하는 구체적인 보안 통제 방안을 도출하였다. 마지막으로, 매핑된 데이터를 기반으로 보안 강도를 측정함으로써 정보 시스템의 보안 수준을 종합적으로 평가하고 개선 방안을 제시하였다.

본 논문의 구성은 다음과 같다. 제2장에서는 이론적 배경을 다루고, 제3장에서는 연구 방법론을 설명하며, 제4장에서는 보안 강도 측정 방법을 설명한다. 마지막으로 제5장에서는 결론과 향후 연구 방향을 제시한다.

II. Related Theory and Technical Background

1. Information Protection System in the United States

NIST SP 800-53 Rev.5[2]는 국립표준기술연구소(NIST : National Institute of Standards and Technology)에서 작성되었으며, 미국 연방 정보보안 관리법(FISMA)에 근거하여 특별 고시(SP: Special Publication) 형태로 제공되는 출판물이다[3]. NIST SP 800-53은 미국 정부 기관이 정보보안 시스템을 설계, 구현 및 관리하는 방법에 대한 표준과 지침을 제공한다.

NIST SP 800-53의 보안 통제항목들은 크게 관리적, 운영적, 기술적 측면으로 분류된다[4]. 관리적 통제는 조직의 보안 활동을 관리하고 감독하는 항목들로, 조직의 전반적인 보안 체계를 구축하고 유지하는 데 중점을 둔다. 운영적 통제는 시스템의 일상적인 운영 과정에서 발생하는 보안 활동에 관하여 다루는 항목들로, 시스템의 안전성을 지속적으로 보장하고 운영 도중 발생할 수 있는 보안 위협에 대응하는 데 목적이 있다. 마지막으로, 기술적 통제는 시스템 내에서 기술적으로 구현되는 보안 수단들에 관한 항목들이다.

앞서 언급한 분류 기준에 따라 NIST SP 800-53 Rev.5의 보안 통제항목들은 Table 1과 같다. 접근통제 항목들은 각각 하위 통제항목들을 포함한다.

2. CCI and SRG/STIG

2.1 CCI(Control Correlation Identifier)[5]

CCI는 DISA FSO가 관리하는 사양[6]으로, 정보보안 통제항목이나 개별적이고 실행 가능한 문장에 대해 표준 식별자와 설명을 제공한다[7]. CCI는 NIST SP 800-53이나 DoDI 8500.2와 같은 고위급 정책을 저수준 기술 구현으로 변환할 수 있도록 설계되어 있다. CCI는 고위급 정책 프레임워크에서 표현된 보안 요구사항을 실행 가능한 단일 진술로 구체화하고, 해당 보안 통제항목의 준수 여부를 평가하는 데 필요한 저수준 보안 설정과 명확하게 연결한다.

미국 국방부 기관의 사이트에서는 CCI 목록을 xml, html의 형태로 제공하는데, 각 CCI ID는 출판 날짜, 기여자, 정의, 유형, 레퍼런스를 포함한다. 이때 하나의 CCI ID는 NIST SP 800-53의 세부적인 보안 통제항목들과 직접 매핑되어 있다. Table 2는 CCI ID와 NIST SP 800-53의 보안 통제항목이 어떻게 매핑되어 있는지를 보여준다. 예를 들어, CCI-000001은 NIST SP 800-53의 AC-1 항목과 매핑된다.

Table 1. List of Security Controls in NIST SP 800-53 Rev.5

Category	Identifier	Number of Controls	Description
Management			
Risk Assessment	RA	9	Activities for assessing and managing security risks in an organization
Planning	PL	8	Includes establishing and managing security plans
System and Services Acquisition	SA	16	Procedures for introducing secure systems and services
Supply Chain Risk Management	SR	12	Assessment and management of security risks across the supply chain
Operational			
Awareness and Training	AT	5	Enhancing security awareness through employee education and training
Contingency Planning	CP	12	Ensuring continuous system operation during emergencies
Incident Response	IR	9	Rapid response procedures in case of a security incident
Maintenance	MA	7	Routine maintenance of information systems
Mdeia Protection	MP	8	Procedures for protecting data storage media
Physical and Environmental Protection	PE	22	Physical environment protection of information systems
Program Management	PM	32	Overall management of security programs
Personnel Security Family	PS	9	Management of staff security qualifications and activities
Technical			
Access Control	AC	23	Access control and management of system resources
Audit and Accountability	AU	15	System usage logging and monitoring
Configuration Management	CM	14	Change management and control of system components
Assessment Authorization and Monitoring	IA	8	Continuous assessment and monitoring of system security status
Identification and Authentication	CA	12	Technical means for identifying and authenticating users or devices
PII Processing and Transparency	PT	8	Ensuring transparency and protection in personal data processing
System and Communications Protection	SC	47	Communication protection between networks and systems
System and Information Integrity	SI	22	Technical methods for maintaining system integrity

Table 2. Example of CCI List

cci_id	publish date	definition	references
CCI-000001	2009-05-13	The organization develops an access control policy that addresses purpose, scope, roles, responsibilities, management commitment, coordination among organizational entities, and compliance.	NIST SP 800-53 AC-1 a NIST SP 800-53A, AC-1.1 (i and ii) NIST SP 800-53 Rev. 4, AC-1 a 1

2.2 SRG and STIG

SRG(Security Requirements Guides)와 STIG(Security Technical Implementation Guides)는 미국 국방부(DoD)의 IT 시스템과 네트워크를 보호하기 위해 개발된 보안 지침이다. 이 두 지침은 서로 보완적이며, 다음과 같은 특징을 가진다. SRG는 다양한 기술 및 제품의 특성을 고려하여 CCI로 그룹화된 특정 기술 분야에 대한 보안 요구사항의 집합이다[8]. SRG는 국방부 기준선에 포함되었는지와 관계없이 상위 수준에서 적용 가능하며, 특정 제품이나 기술에 대한 구체적인 보안 요구사항을 제공한다. 또한, 보안 통제 식별자(CCI)와 보안 기술 구현 가이드(STIG) 사이의 중간 단계 역할을 한다.

SRG를 기반으로 한 STIG[9]는 국방 정보시스템국(DISA)이 개발한 설정 표준으로, 특정 제품의 보안 요구사항을 구체적인 기술 지침으로 변환하였다. 즉, 보안 요구사항을 실제 시스템과 네트워크에서 적용할 수 있도록 한다. 정리하면 SRG는 보안 요구사항을 정의하는 역할을 하고, STIG는 요구사항을 구체적으로 구현하는 역할을 한다. SRG는 CCI를 통해 정의된 보안 요구사항을 그룹화하고, STIG는 CCI의 요구사항들을 구체적인 보안 설정으로 구현한다.

3. CVE and OVAL Benchmark

3.1 CVE(Common Vulnerabilities and Exposures)

CVE는 공개적으로 알려진 컴퓨터 보안 취약점을 식별하기 위한 표준 체계다. 보안 취약점에 CVE ID 번호를 할당하고 심각도를 측정함으로써 보안 전문가들이 취약점에 우선순위를 지정하고 해결할 수 있도록 지원한다.

CVE ID는 CNA(CVE Numbering Authority) 기관에서 할당한 고유한 영숫자 식별자로써[10], 재품에 영향을 미치는 취약성에 대한 번호를 부여한다. CVE ID의 형식은 'CVE prefix + Year + Arbitrary Digits(또는 Sequence Number)'로 구성된다. 여기서 Year는 CVE ID가 예약된 연도나 취약성이 공개된 연도를 나타내며, 취약점이 발견된 시기를 직접적으로 나타내지는 않는다. Arbitrary Digits(또는 Sequence Number)는 4자리 이상의 숫자로 구성될 수 있으며, 자릿수에는 제한이 없다. 이런 CVE ID 형식은 매년 발생하는 많은 취약점을 체계적으로 관리하기 위해 설계되었다.

Table 3. Structure of CVE Record

Field	Description
CVE ID	CVE ID with "Arbitrary Digits (or Sequence Number)" consisting of 4 or more digits (e.g., "CVE-1999-0067", "CVE-2021-7654321")
Description	A brief explanation of the security vulnerability
Product Status	Products and versions affected by the vulnerability
References	All relevant references (e.g., vulnerability reports and advisories)

CVE 레코드(CVE Record)는 CNA에서 제공하는 CVE ID와 관련된 취약성에 대한 설명 데이터이다. 이 데이터는 사람과 기계가 읽을 수 있는 형식으로 제공된다. CVE 레코드는 Table 3의 요소로 구성된다.

3.2 OVAL(Open Vulnerability and Assessment Language)[11]

OVAL(Open Vulnerability and Assessment Language)은 컴퓨터 시스템의 상태를 평가하고 취약점을 보고하는 과정을 표준화하기 위해 만들어진 국제 정보보안 커뮤니티 표준이다. 이 표준은 시스템 관리자가 로컬 시스템의 특성과 설정 정보를 기반으로 취약점을 식별하고 평가할 수 있도록 한다. OVAL 언어는 시스템의 설정 정보를 수집하여 보안 정책의 준수 여부를 검증하는 데 사용된다.

OVAL을 이용한 시스템 평가 프로세스는 크게 세 단계로 구분된다. 첫째, 평가 대상 시스템의 설정 파일과 경로를 확인한다. 이후, 보안 정책을 참고하여 수집할 객체와 해당 객체가 준수해야 하는 상태를 OVAL 정의(OVAL Definition) 파일로 작성한다. 여기서 시스템 객체는 파일, 디렉터리 등과 같이 보안 평가에 필요한 다양한 시스템 자원을 포함한다. OVAL 정의 파일은 시스템 객체가 보안 정책을 준수하는지 기술적으로 명시한 문서이다. 둘째, 작성된 OVAL 파일은 시스템 스캐너에 입력된다. 시스템 스캐너는 정의된 OVAL 정의 파일을 바탕으로 시스템 객체를 수집하고, 이를 OVAL 시스템 특성(OVAL System Characteristics) 파일로 변환한다. 마지막으로, 이렇게 수집된 시스템 특성 정보는 사전에 작성된 OVAL 정의 파일과 비교되어 평가가 이루어진다. 평가 결과는 OVAL 결과(OVAL Results) 파일로 제공되며, 결과 파일을 통해 시스템의 보안 상태 및 취약성 여부를 검토할 수 있다.

(1) OVAL Class

OVAL 정의는 시스템 보안 정책 및 취약점을 점검하기 위한 목록으로, OVAL 클래스는 다양한 보안 평가 목적을 구분한다. OVAL 5.11 기준으로 컴플라이언스, 인벤토리, 패치, 취약점, 기타의 다섯 가지 클래스가 있다. 컴플라이언스 클래스는 시스템이 정책을 준수하는지 확인하며, 인벤토리 클래스는 특정 소프트웨어의 설치 여부를 점검한다. 패치 클래스는 취약점에 대한 패치 적용 여부를 확인하고, 취약점 클래스는 시스템의 취약성을 평가한다. 마지막으로 기타 클래스는 다른 정의에 속하지 않는 다양한 시스템 상태를 평가한다.

(2) Structure of OVAL Definition for Vulnerability/Patch Class

OVAL 정의 파일은 Fig. 1.과 같이 크게 생성 정보(generator), 정의(definitions), 테스트(tests), 객체(objects), 상태(states), 그리고 변수(variables)의 여섯 가지 주요 구성요소로 이뤄진다. 생성 정보(generator)는 정의 파일의 생성 정보를 포함하며, 작성자, 작성 날짜 및 파일의 버전 정보 등을 담고 있다. 정의(definitions)는 시스템이 준수해야 할 조건이나 특정 취약점의 존재 여부를 명시하는 보안 점검목록을 포함한다. 테스트(tests)는 시스템 객체가 특정 조건을 만족하는지 확인하기 위해 수행되는 테스트 방법을 정의하며, 각 테스트는 관련된 객체(objects)와 상태(states)를 참조한다.

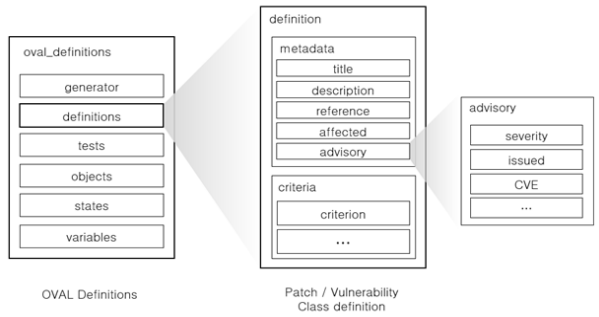

Fig. 1. Structure of OVAL Definition for Vulnerability/Patch Class

객체(objects)는 테스트할 시스템 객체를 정의하며 파일, 디렉터리, 프로세스 등 다양한 시스템 구성 요소를 포함한다. 상태(states)는 시스템 객체가 만족해야 하는 상태를 명시하며, 특정 파일이 존재하는지 또는 수집된 객체의 값이 보안 정책과 맞는지 등을 확인한다. 마지막으로, 변수(variables)는 여러 정의에서 공통적으로 사용할 수 있는 재사용 가능한 변수를 정의한다.

특히 취약성 클래스와 패치 클래스의 경우, 정의(definition) 요소는 다른 클래스의 OVAL 정의와는 달리 시스템이 취약한 상태인지 평가하는 데 중점을 두고 작성된다. 따라서 보안 점검의 기준을 명확히 하기 위해 다양한 메타데이터(metadata)를 포함하며, 다른 클래스와의 차별화된 요소인 권고 사항(advisory) 부분이 강조된다.

메타데이터(metadata) 부분은 정의 파일의 제목, 참조, 설명 등을 포함하며, 취약성/패치 클래스 OVAL 정의의 경우 권고 사항(advisory) 요소를 포함한다. 권고 사항(advisory)은 정의된 취약성이나 패치에 대한 권고 사항을 제공하는데, 여기에는 심각도 수준(severity), 발행 날짜(issued), CVE 정보(CVE) 등이 포함된다. 심각도 수준(severity)은 해당 취약성이나 패치의 중요도를 나타낸다. 발행 날짜(issued)는 취약성이나 패치가 처음 보고되거나 발행된 날짜를 명시한다. CVE 요소는 특정 CVE 취약점에 대한 고유 식별 정보를 제공한다. 이는 취약점의 세부 사항과 영향을 받는 구성요소에 대한 정보를 포함한다.

4. Previous Research

기존 연구는 주로 NIST SP 800-53과 MITRE ATT&CK 간의 연관성을 분석하거나 CVE와 다른 보안 프레임워크 간의 관계를 자동화하는 데 중점을 두고 있다. 하지만 NIST SP 800-53과 CVE 간의 직접 매핑을 다룬 연구는 거의 없어 보안 관리자들이 통제의 중요성과 취약점 심각도를 평가하는 데 어려움이 있다.

Marchiori et al.[12]는 NIST SP 800-53과 MITRE ATT&CK 간의 연관성을 분석하여 보안 통제 실패가 전체 보안에 미치는 영향을 평가하기 위해 퍼지 논리와 사이버 위험 평가 모델을 사용했다. 보안 통제와 공격 기법 간의 관계를 분석하여 위험 값(RVC)을 할당하고, 보안 통제 종속 그래프(SCDG)를 설계했다. 평균 F1 점수는 0.3 이하로 최대 0.57을 기록하여 기존 방식의 한계를 나타냈으며, AI 모델을 통한 새로운 연구가 필요함을 시사한다.

식별자와 CVE 간 매핑을 추진한 연구들도 존재한다. I. Branescu et al.[13]는 CVE와 MITRE ATT&CK 전술을 매핑하는 연구를 수행했다. Transformer 기반 모델(SecBERT, CyBERT, SecRoBERT 등)을 사용하여 14가지 전술에 매핑하였고, 총 9985개의 항목을 포함했다. SecRoBERTa 모델이 가장 높은 F1 점수(78.88%)를 기록했다. 이 연구는 Transformer 모델이 CVE 설명을 효과적으로 분석할 수 있음을 보여주었다.

Haddad et al.[14]는 CVE 설명을 MITRE CWE 약점으로 매핑하는 연구를 수행했다. 사전 학습된 모델(SBERT, rankT5, RoBERTa, BERT 등)을 보안 도메인에 특화된 데이터로 추가 학습하여 CVE와 CWE 설명 간 유사성을 평가했다. RankT5 모델이 유사도 기반 매핑에서 F1 점수 78.5%로 가장 높은 성능을 기록했다.

기존 연구들이 탐구하지 않은 영역인 NIST SP 800-53과 CVE 간의 직접 매핑을 다룸으로써 본 연구는 보안 평가의 정확성과 실효성을 높일 것으로 기대된다.

III. AI-Based Mapping Process

본 연구는 NIST SP 800-53과 CVE 간의 상호 연관성을 분석하여 보안 평가의 우선순위를 설정하는 데 목적을 둔다. 이를 위해 STIG와 OVAL 파일 간의 매핑을 통해 CVE와 STIG 간의 매핑을 수행하였다. 매핑 과정에 대한 개요는 Fig. 2.와 같다.

첫째, NIST SP 800-53, CCI, SRG, STIG 간에는 각 보안 통제항목이 일관성 있게 연결되므로 바로 연결 작업을 수행한다. 미 국방부 사이트에서 제공하는 CCI 목록에서는 각 CCI 항목마다 NIST SP 800-53의 보안 통제 항목번호를 직접 참조한다. 또한, SRG 항목은 ident 태그에서 CCI 항목번호를 직접 참조하고 version 태그에서 STIG 항목번호를 직접 참조한다. 따라서 NIST SP 800-53과 CCI, SRG, STIG 간 요구사항은 서로 상충하지 않으므로 바로 연결 작업을 수행할 수 있다.

둘째, OVAL 파일을 이용하여 STIG와 CVE 간의 매핑을 수행했다. 취약성 클래스 항목을 저장한 OVAL 파일은 CVE ID를 직접 참조하여 취약점을 식별하고 상세한 정보

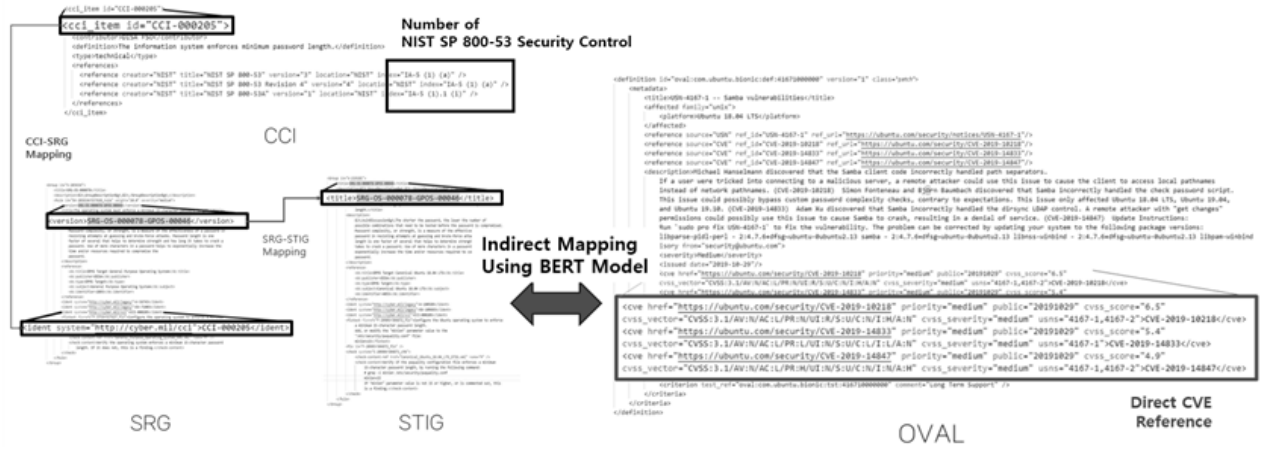

Fig. 2. Mapping Process between NIST SP 800-53 and CVE

를 제공한다. 이를 활용하여 STIG와 OVAL 정의 항목을 매핑함으로써 CVE와 STIG가 연결될 수 있도록 한다.

STIG 항목은 주로 시스템 구성요소의 설정과 보안 요구사항을 규정하지만, OVAL 정의는 특정 취약점의 세부 사항과 이를 악용했을 때 발생할 수 있는 문제점에 관해 기술한다. STIG와 OVAL 간에는 보안 요구사항이 일치하지 않거나 상충될 수 있는 경우가 존재한다. 이를 해결하기 위해 사전 정의된 매핑 기준을 설정하였다. 일부 파일에 대해서는 이 기준을 따라 간접 수동 매핑을 수행하였으며, 수동 매핑은 상충 가능성을 최소화하고 매핑의 일관성을 유지하기 위한 필수적인 과정이었다. 나머지 항목에 대해서는 사전 학습된 인공지능 모델(SecBERT, CyBERT, RankT5)을 이용해 자동 매핑을 효율적으로 수행하였다. STIG의 설명(description)과 OVAL의 설명(description) 간의 유사도를 비교하기 위해 BERT 모델을 사용한 이유는 BERT 모델이 텍스트의 문맥적 의미를 효과적으로 캡처할 수 있기 때문이다. 이렇게 매핑을 수행함으로써 NIST SP 800-53과 CVE가 매핑될 수 있도록 하였다.

1. Experimental Environment and Dataset

본 연구는 STIG 파일이 충분히 제공되고 점유율이 높은 우분투 환경(18.04 LTS, 20.04 LTS, 22.04 LTS)에 한정하여 진행되었다. 실험은 Ubuntu 22.04.3 LTS 운영체제에서 수행되었으며, 하드웨어로는 NVIDIA Tesla T4 GPU를 사용하였다. 소프트웨어 환경은 Python 3.10.12 버전을 기반으로 하였으며, 주요 라이브러리의 버전은 Table 4에 나타냈다.

본 연구에서는 OVAL 데이터셋과 STIG 데이터셋을 수집하여 실험을 진행하였다. OVAL 파일의 경우, 우분투[16]의 전용 레포지토리에서 다운로드하여 사용하였다. 레드햇과 우분투는 각 운영체제에 맞는 보안 평가를 위해 자체적으로 OVAL 정의 파일을 제공하며, 이를 통해 시스템 구성 요소를 검사하고 취약점을 평가할 수 있다. 이 OVAL 파일들은 최신 보안 정책과 취약점을 반영하여 정기적으로 업데이트된다.

STIG 파일은 NCP(National Checklist Program) 체크리스트[17]에서 다운로드하여 사용하였다. NCP는 미국 국립 표준 기술 연구소(NIST)에서 운영하는 프로그램으로, 다양한 시스템과 애플리케이션에 대한 보안 구성 기준을 제공한다. NCP 체크리스트에서 제공하는 STIG 파일은 미국 국방부(DISA)가 주도하여 작성된 것으로, 높은 신뢰성과 보안성을 보장한다. Table 5는 수집한 OVAL. STIG 데이터셋의 전체 목록을 보여준다.

Table 4. List of OVAL and STIG Datasets

File Type	File Name	Number of Items
OVAL [15]	com.ubuntu.bionic.usn.oval.xml	2140
	com.ubuntu.focal.usn.oval.xml	1732
	com.ubuntu.jammy.usn.oval.xml	914
STIG [17]	U_CAN_Ubuntu_18-04_LTS_STIG_V2R14_Manual-xccdf.xml	176
	U_CAN_Ubuntu_20-04_LTS_STIG_V1R12_Manual-xccdf.xml	167
	U_CAN_Ubuntu_22-04_LTS_STIG_V1R1_Manual-xccdf.xml	184

Table 5. Library Versions Used in This Study

Library Name	Version
PyTorch	2.3.0 (CUDA 12.1)
Transformers	4.41.2
Pandas	2.0.3
Numpy	1.25.2
Scikit-learn	1.2.2

2. Data Preprocessing of STIG and OVAL Files

2.1 Manual Mapping of Training Data

인공지능 모델에 넣을 데이터셋을 생성하기 위해, 본 연구에서는 일부 OVAL 파일과 STIG 파일에 대해 수동 매핑을 진행했다. 수동 매핑에 사용된 데이터는 'com.ubuntu.bionic.usn.oval.xml' OVAL 파일과 'U_CAN_Ubuntu_18-04_LTS_STIG_V2R14_Manual-xccdf.xml' STIG 파일이며, 나머지 파일은 자동 매핑을 수행하였다.

NIST SP 800-53, CCI, SRG/STIG와 OVAL 간의 매핑 작업은 보안 평가의 우선순위를 설정하고 취약점을 효과적으로 관리하기 위해 중요한 과정이다. 그러나 이 과정에서 STIG 항목과 OVAL 정의 간의 연관성을 명확히 파악하는 데 어려움이 있다. STIG 항목은 주로 시스템 구성요소의 설정과 보안 요구사항을 규정하는 반면, OVAL 정의는 특정 취약점의 세부 사항과 이를 악용했을 시 발생할 수 있는 문제점에 관한 기술하기 때문이다. 이러한 차이로 인해 두 항목 간의 명확한 매핑 기준을 설정하는 것이 중요하다.

기존 연구에서의 매핑 기준은 주로 두 항목 간의 직접적인 연관성에 초점을 맞추었으나, 이를 본 연구에 적용하는 것은 불가능하였다. 예를 들어, STIG 항목이 시스템 비밀번호 설정과 같은 기본적인 보안 설정을 다루는 경우, 이러한 설정이 특정 취약점과 직접적으로 연결되지 않는 경우가 많다. 이러한 상황에서 두 항목 간의 간접적 관계를 인정하고, 보안 목표와 적용 범위 등을 고려하여 매핑하는 기준이 필요하다. 따라서, 유연하고 포괄적인 매핑 기준을 설정하여 STIG 항목과 OVAL 정의 간의 관계를 명확히 하였다. NIST SP 800-53, STIG, CVE, OVAL 간의 매핑 기준을 다음의 기준으로 설정하였다.

- **보호 대상 및 목적 일치(PA : Protected Assets and Objectives)** : STIG 항목이 보호하려는 대상과 OVAL 정의가 다루는 취약점이 동일한 시스템 구성요소와 관련이 있는지 확인
- **보안 목표 및 의도 일치(SG : Security Goals and Intentions)** : STIG 항목의 보안 목표와 OVAL 정의의 취약점 설명에서 공통된 보안 목표 또는 의도 확인
- **적용 범위 일치(SM : Scope Match)** : TIG 항목과 OVAL 정의가 동일한 환경(예: 운영체제 버전, 네트워크 설정 등)에 적용되는지 확인
- **보안 설정과 취약점의 간접 관계(IR : Indirect Relationship between STIG and OVAL)** : STIG 항목이 요구하는 보안 설정이 OVAL 정의에서 설명하는 취약점과 간접적으로 관련이 있는지 평가

앞서 설명한 기준을 바탕으로 STIG 항목과 OVAL 정의를 매핑하여 체계적이고 일관된 보안 평가를 수행할 수 있도록 하였다. 또한, 객관적인 평가를 위해 Table 6~9의 매핑 점수표를 도입하였다.

Table 6. Protected Assets and Objectives[PA] Scores (Max 10 Points)

Score	Score Description
10	Protects the same system component for both items
7-9	Protects similar components
4-6	Protects components with low relevance
0-3	Protects entirely unrelated components

Table 7. Security Goals and Intentions[SG] Scores (Max 10 Points)

Score	Score Description
10	Achieves the same security goal for both items
7-9	Achieves similar security goals
4-6	Achieves security goals with low relevance
0-3	Achieves entirely different security goals

Table 8. Scope Match[SM] Scores (Max 5 Points)

Score	Score Description
5	Applied to the same environment
3-4	Applied to a similar environment
1-2	Applied to an environment with low relevance
0	Applied to an entirely different environment

Table 9. Indirect Relationship between STIG and OVAL[IR] Scores (Max 5 Points)

Score	Score Description
5	Security configuration directly contributes to vulnerability management
3-4	Security configuration indirectly contributes to vulnerability management
1-2	Low relevance
0	No relevance at all

'보호 대상 및 목적 일치', '보안 목표 및 의도 일치' 기준에 대하여 타 기준의 2배수로 점수를 적용한 이유는 이 두 항목이 STIG 항목과 OVAL 정의 간의 가장 근본적이고 중요한 연관성을 나타내기 때문이다. '보호 대상 및 목적 일치' 기준은 두 항목이 동일한 시스템 구성 요소를 보호하는지를 평가함으로써 가장 기본적인 수준에서의 연관성을 확인한다. '보안 목표 및 의도 일치' 기준은 두 항목이 동일한 보안 목표를 달성하는지를 평가함으로써 보안

설정의 궁극적인 목적과 방향성을 파악할 수 있게 한다. 따라서, 이 두 항목은 매핑의 타당성을 평가하는 데 있어 가장 중요한 요소로 간주되어 배점이 높게 설정되었다. 이 네 기준에 대한 총점 계산 방법은 수식 4와 같다.

$$Total\ Score(TS) = PA + SC + SM + IR \quad (4)$$

하나의 STIG 항목에 대해 각 매핑 점수를 매긴 OVAL 항목들 중 총점 25점 이상을 획득한 OVAL 항목만을 유효한 매핑으로 간주하여, 보다 객관적이고 신뢰성 있는 보안 평가를 수행하였다. 25점 이상을 획득한 항목은 보호 대상 및 보안 목표의 높은 일치율을 보이므로 매핑을 수행하였다. 반면, 20점 이상 24점 이하의 항목은 동일한 시스템 구성요소를 보호하지 않는 경우가 대다수였기 때문에 연관성이 낮아 매핑을 수행하지 않았다.

2.2 Data Preprocessing

(1) Key Information Extraction

OVAL 파일은 보안 취약점에 대한 상세한 정보를 제공하며, 특히 제목(title)과 설명(description) 필드를 통해 이러한 정보를 서술한다. 설명(description) 필드는 주로 취약점의 원인과 영향을 설명하는 내용을 포함하고 있으며, 이는 보안 평가 및 매핑 작업에 있어 중요한 데이터 소스가 된다. OVAL 파일의 설명 필드는 Table 10과 같이 보통 취약점에 대한 다음과 같은 정보를 포함한다.

STIG 항목과 효과적으로 매핑하기 위해 '취약점 설명', '취약점의 영향'을 추출하였으며, 근거는 다음과 같다. 첫째, 취약점 설명은 보안 취약점이 발생하는 구체적인 이유를 제공하여 STIG 항목과의 매핑을 정확하게 할 수 있게 한다. 취약점 원인은 보안 설정의 부재나 잘못된 설정이 어떤 취약점을 초래하는지를 명확하게 이해할 수 있게 한다. 둘째, 취약점의 영향은 취약점의 발생에 따라 영향을 받는 시스템 구성 요소에 대한 정보와 공격종류에 대한 담고 있으므로 추출하였다. 특히 취약점 설명은 취약점이 발생하는 상황에 대한 맥락을 제공하여, AI 모델이 더 정확하게 매핑할 수 있도록 돕는다.

(2) Data Preprocessing

데이터는 STIG 항목과 OVAL 정의를 포함하는 CSV 파일로부터 로드되었다. 데이터셋은 여러 가지 속성으로 구성되어 있으며, 주요 속성은 Table 11과 이 'Group Title', 'Rule Title + Description', 'OVAL ID', 'oval_text', 'is_mapped'로 구성된다. 'Group Title'은 STIG 항목의 그룹 제목을 나타내며, 예를 들어 "V-219147"과 같은 형식으로 제공된다. 이 속성은 STIG 항목을 분류하는 데 사용된다. 'Rule Title + Description'은 STIG 항목의 규칙 제목과 설명을 포함하고 있으며, STIG 항목의 구체적인 내용을 제공한다. 이 속성은 모델의 주요 입력 데이터 중 하나로 사용된다. 'OVAL ID'는 OVAL 정의의 고유 식별자를 나타낸다. 'oval_text'는 OVAL 정의의 구체적인 내용을 설명하는 텍스트를 포함하며, 이 속성 역시 모델의 주요 입력 데이터

Table 10. Structure of Content in the Description Field of OVAL Definition

Component	Description	Example
Vulnerability Description	A general explanation of the situation in which the vulnerability occurs	Alex Nichols and Jakob Hirsch discovered that the Apache HTTP Server mod_authnz_ldap module incorrectly handled missing charset encoding headers.
Vulnerability Impact	The potential impact of the vulnerability on the system	A remote attacker could possibly use this issue to cause the server to crash, resulting in a denial of service.
CVE Number	The CVE number assigned to the vulnerability	(CVE-2017-15710)
Update Instructions and Package Version	The update method and software version required to fix the vulnerability	Run sudo pro fix USN-3627-2 to fix the vulnerability. The problem can be corrected by updating your system to the following package versions: apache2-data - 2.4.29-1

Table 11. Example of Dataset Composition

Group Title	Rule Title + Description	OVAL ID	oval_text	is_mapped
V-219147	Ubuntu operating systems booted with a BIOS must require...	oval:com.ubuntu.bionic:def:44321000000	GRUB 2 vulnerabilities Jesse Michael and Micke...	True
V-219147	Ubuntu operating systems booted with a BIOS must require...	oval:com.ubuntu.bionic:def:44322000000	GRUB2 regression Jesse Michael and Mickey Shka...	True
V-219147	Ubuntu operating systems booted with a BIOS must require...	oval:com.ubuntu.bionic:def:44321000000	GRUB 2 vulnerabilities Jesse Michael and Micke...	True

로 사용된다. 'is_mapped'는 STIG 항목과 OVAL 정의가 매핑되었는지 여부를 나타내는 Boolean 값이다. True 또는 False 값으로 구성되어 있으며, 모델의 학습 레이블로 사용된다.

전처리 과정에서는 BERT 모델이 문맥적 의미를 효과적으로 학습할 수 있도록 최소한의 전처리만 수행하였다. 텍스트 데이터는 공백을 제거하고 정리하는 작업만 진행하였다. 특수 문자와 기호도 정리되었지만, 알파벳 소문자 변환, 숫자 및 특수 문자 제거, 불용어 제거, 표제어 추출 등의 추가 전처리는 BERT 모델의 문맥 학습에 방해가 될 수 있어 배제하였다.

본 연구에서는 Table 11의 속성 중 'Rule Title + Description', 'oval_text', 'is_mapped' 속성을 모델 입력 데이터로 사용하였다. 'Rule Title + Description'과 'oval_text'는 모델의 텍스트 입력으로 사용되었고, 'is_mapped'는 모델의 학습 레이블로 사용되었다. 데이터 전처리 과정에서 'Rule Title + Description'과 'oval_text' 속성을 [SEP] 토큰으로 구분하여 결합하였다. 예를 들어, 각 데이터 포인트에 대해 "Ubuntu operating systems booted with a BIOS must require... [SEP] GRUB 2 vulnerabilities Jesse Michael and Micke..."와 같은 형식으로 결합하였다.

결합된 텍스트 데이터는 BERT 모델의 토크나이저를 사용하여 토큰화되었다. 토큰화 과정에서는 텍스트를 단어 또는 단어 조각으로 분리하고, 이를 정수 인덱스로 변환하여 BERT 모델이 이해할 수 있는 형태로 변환하였다. 이 과정에서 텍스트의 최대 길이를 512로 설정하고, 이보다 긴 텍스트는 자르고 짧은 텍스트는 패딩하여 고정된 길이를 유지하도록 하였다.

또한, BERT 모델에서 문장을 처리할 때 어텐션 마스크를 생성하여 실제 단어가 있는 위치와 패딩된 위치를 구분하였다. 어텐션 마스크는 실제 단어가 있는 위치에는 1을, 패딩된 위치에는 0을 할당하여 모델이 실제 단어에만 주의를 기울이도록 하였다. 이로써 모델이 패딩된 부분을 무시하고 의미 있는 단어들에 집중할 수 있게 되었다. 토큰화된 입력 데이터와 어텐션 마스크는 PyTorch 텐서로 변환되어 모델의 입력으로 사용되었다. 또한, 'is_mapped' 속성은 정수 텐서로 변환되어 학습 레이블로 사용되었다.

데이터셋을 생성하는 과정에서 총 65개의 STIG와 215개의 OVAL을 매핑하여 총 215개의 매핑된 데이터를 생성하였다. 또한, 매핑된 데이터의 3배수인 645개의 매핑되지 않은 데이터를 추가로 결합하여 총 860개의 데이터셋을 만들었다. 이러한 과정에서 자연스럽게 클래스 불균형이 발생하였다. 임의로 클래스 불균형을 발생시킨 이유는 수동 매핑한 데이터셋이 상대적으로 적기 때문에 선택한 방식이다. 발생된 클래스 불균형은 후술되듯 역빈도 가중 손실(Inverse Frequency Weighted Loss) 기법을 이용하여 해결하였다.

최종적으로, 'Rule Title + Description'과 'oval_text'를 [SEP] 토큰으로 구분하여 결합한 텍스트 데이터는 모델의 주요 입력으로 사용되었고, 'is_mapped' 값은 모델의 학습 레이블로 사용되었다. 또한 데이터셋을 학습과 테스트 데이터로 나누는 과정에서 데이터의 80%는 학습용으로, 나머지 20%는 테스트용으로 사용되어 학습 데이터를 BERT 모델에 입력했다.

2.3 Automated Mapping Process

본 연구에서는 STIG와 OVAL 간의 매핑을 수행하기 위해 인공지능 모델을 사용하였다. 사용된 모델들은 SecBERT, CyBERT, 그리고 RankT5이며, 인공지능 모델 최적화 과정 및 매핑 프로세스는 다음과 같다.

(1) AI Model Optimization

본 연구에서는 STIG(Secure Technical Implementation Guide) 항목과 OVAL(Open Vulnerability and Assessment Language) 정의를 매핑하는 과정에서 SecBERT와 CyBERT 모델을 활용하였다. 이 과정에서 모델의 성능을 향상시키기 위해 하이퍼파라미터 최적화를 수행하였다.

SecBERT 모델은 사전 학습된 jackaduma/SecBERT 모델을 기반으로 하였으며, Hugging Face의 BertForSequenceClassification 클래스를 사용하여 구현되었다. 모델의 출력 레이어는 특정 작업에 맞게 조정되었다. CyBERT 모델은 사전 학습된 SynamicTechnologies/CYBERT 모델을 기반으로 하였으며, AutoModelForSequenceClassification 클래스를 사용하여 구현되었다.

앞서 발생한 클래스 불균형 문제를 해결하기 위해, 타 논문에서 효과를 입증한 역빈도 가중 손실(Inverse Frequency Weighted Loss) 기법[18]을 이용하였다. 역빈도 가중 손실은 각 클래스의 빈도수의 역수를 가중치로 사용하여, 손실함수를 조정하는 방법으로 소수 클래스에 대해 더 잘 학습하도록 한 방법이다. 먼저 데이터셋에서 'np.bincount' 함수를 사용하여 라벨 데이터 내에서 각 클래스의 빈도를 집계함으로써 각 클래스의 분포를 파악하였다. 이후 빈도수를 기반으로 클래스 가중치를 계산하였는데, 클래스 가중치는 각 클래스의 빈도수의 역수로 정의된다. 예를 들어, 클래스 0의 빈도수가 50이고 클래스 1의 빈도수가

150인 경우, 클래스 가중치는 각각 1/50과 1/150으로 설정된다. 마지막으로 'torch.nn.CrossEntropyLoss' 함수의 weight 파라미터에 앞서 계산한 클래스 가중치를 전달하여 불균형한 데이터셋에서도 모델이 균형 잡힌 성능을 발휘할 수 있도록 하였고, 또한 하이퍼파라미터 최적화를 Optuna 라이브러리를 사용하여 F1 스코어를 최대화하는 방향으로 최적화를 수행하였다.

Table 12와 같이 SecBERT 모델의 최적 하이퍼파라미터는 드롭아웃 비율 0.2, 학습률 9.7324e-05, 배치 크기 8로 설정되었다. CyBERT 모델의 경우, 최적 하이퍼파라미터는 드롭아웃 비율 0.2, 학습률 2.5106e-05, 배치 크기 32로 결정되었다.

Table 12. Model Optimization

Model	dropout_ rate	learning_ rate	batch_ size	epoch
SecBERT	0.2	9.7324e-05	8	3
CyBERT	0.3	3.4515e-05	32	5

(2) Model Performance Evaluation

파인튜닝을 수행하지 않은 SecBERT, CyBERT, RankT5의 성능 및 파인튜닝을 수행한 모델들의 성능을 비교한 결과는 다음과 같다. 파인튜닝 전 SecBERT 모델은 정확도(Accuracy) 32.55%, 정밀도(Precision) 27.15%, 재현율(Recall) 95.67%, F1 점수 42.30%로 전반적으로 낮은 성능을 보였다. 특히 정밀도가 낮은 것은 모델이 실제 Positive 샘플을 Negative로 잘못 예측하는 경우가 많았음을 의미한다. 하지만 파인튜닝 후에는 정확도 96.27%, 정밀도 94.97%, 재현율 86.05%, F1 점수 92.50%로 모든 지표에서 큰 폭으로 향상되었다. 이는 파인튜닝을 통해 모델이 데이터 특성을 더 잘 학습하고 일반화 성능을 높였음을 시사한다.

파인튜닝 전 CyBERT 모델은 정확도 25.84%, 정밀도 25.84%, 재현율 100%, F1 점수 41.07%로 SecBERT와 유사하게 낮은 성능을 나타냈다. 특히 정확도와 정밀도가 매우 낮은 것은 모델이 Positive 샘플을 제대로 예측하지 못하고 대부분 Negative로 예측했음을 의미한다. 파인튜닝 후에는 정확도 85.71%, 정밀도 71.43%, 재현율 90.00%, F1 점수 79.65%로 성능이 크게 향상되었지만, SecBERT에 비해 개선 폭은 작았다. CyBERT 모델이 데이터 특성을 학습하는 데 어려움을 겪었거나, 파인튜닝 과정에서 최적화가 충분히 이루어지지 않았을 가능성을 시사한다.

마지막으로, RankT5 모델은 파인튜닝을 하지 않았음에도 정확도, 정밀도, 재현율, F1 점수 모두 100%로 완벽한 성능을 보였다. 이에 반면 CyBERT와 SecBERT는 현실적인 성능을 보인 것을 확인하였으며, 이를 통해 RankT5 모델의 결과가 과적합이나 모델의 과도한 특수성에 기인한 것일 수 있음을 시사한다. 따라서, 본 연구에서는 두 번째로 높은 SecBERT 모델의 성능을 최종 결과로 채택하여 매핑을 수행했다.

IV. Measuring Security Strength of the U.S. Information Protection System

1. Definition and Measurement of Security Strength

보안 강도(Security Strength)는 특정 보안 통제가 얼마나 많은 취약점(CVE)과 연관되어 있는지를 기준으로 우선적으로 평가되어야 한다.

보안 강도는 다음과 같은 단계들을 통해 측정된다. 첫째, 보안 통제항목을 추출하고, NIST SP 800-53의 보안 통제항목 중 기술적 항목을 선별한다. 둘째, 데이터 매핑을 수행한다. CCI, SRG, STIG와 CVE 간의 매핑을 통해 데이터 연계를 시도한다. 이 과정에서 앞서 인공지능 모델을 통해 자동으로 매핑한 OVAL 파일의 리스트를 이용하였으며, 참조된 CVE 항목을 연결함으로써 최종적으로 매핑을 수행하였다. 마지막으로, 보안 강도를 평가한다. 매핑된 데이터를 바탕으로 각 보안 통제가 얼마나 많은 CVE와 연결되어 있는지를 분석하여 보안 강도를 평가함으로써 보안 평가의 우선순위를 결정할 수 있다.

보안 강도 측정 결과는 보안 통제가 취약점을 얼마나 효과적으로 관리하는지를 평가하는 데 사용된다. 예를 들어, 높은 수의 CVE와 연결된 보안 통제는 중요한 취약점을 포함하고 있을 가능성이 커, 이러한 통제항목들은 우선적으로 평가하고 개선해야 할 필요가 있다. 보안 강도 평가는 조직의 전반적인 보안 상태를 향상시키고, 시스템 보안을 보다 체계적이고 효과적으로 관리하는 데 기여할 것으로 기대한다.

2. Security Strength Evaluation

미국 정보보호체계 간 관계 매핑을 통해 Fig. 3, Fig. 4와 같이 네트워크 그래프를 얻을 수 있었으며, 보안 통제항목과 CVE 간의 관계를 시각화하였다. 각 노드는 보안

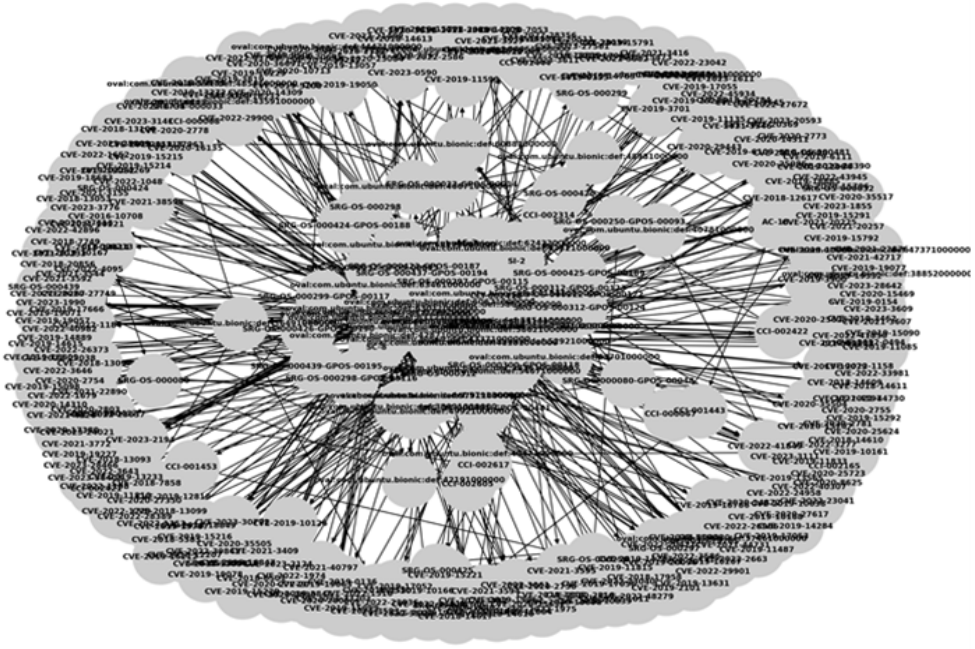

Fig. 3. Network of All Mapped Items

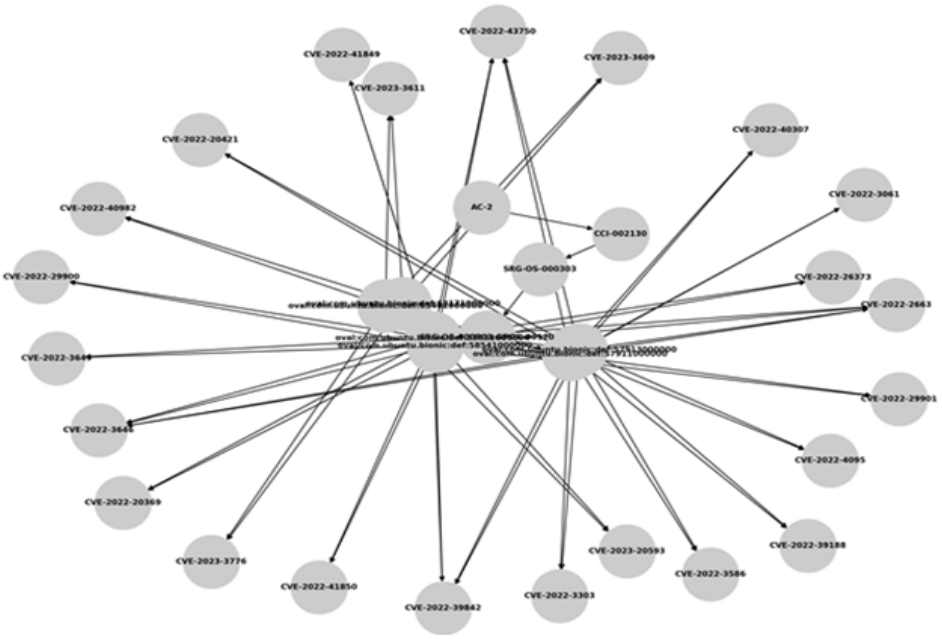

Fig. 4. Relationship Network of Mapped Items (AC-2)

통제항목 또는 CVE를 나타내며, 링크는 이들 간의 매핑을 나타낸다. 예를 들어 'AC-2' 보안 통제항목을 기준으로, Fig. 4.와 같이 OVAL를 통해 여러 CVE와 보안 통제항목을 매핑할 수 있었다. 이 그림에서 'AC-2' 보안 통제 항목은 다수의 CVE와 연관되어 있음을 알 수 있다. 이는 'AC-2' 항목이 중요한 보안 통제임을 의미하며, 이 항목에 대한 보안 강화를 통해 많은 취약점을 효과적으로 관리할 수 있음을 시사한다.

Fig. 5.를 확인한 결과, 우분투 기준으로 AC-17(원격 접근), AC-3(접근통제 강화), AC-6(최소 권한), IA-5(사용자 인증 정보 관리) 등 핵심적인 보안 통제항목들의 CVE 비중이 높은 것을 확인할 수 있었다. 특히 최신 운영체제에서 발생된 취약점들을 포함하고 있어 우선적으로 개선할 필요가 있음을 나타낸다.

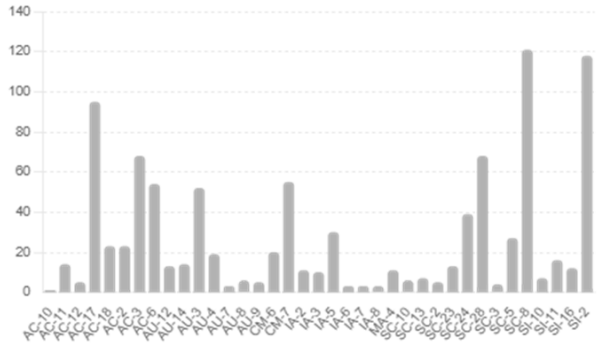

Fig. 5. Distribution of CVE Items Mapped to Sub-Control Items

3. Discussion and Limitations

본 연구에서는 NIST SP 800-53 보안 통제항목과 CVE 간의 관계를 분석하여 보안 강도 네트워크를 구축하였다. 그러나 연구 결과에서 나타난 CVE 분포의 불균형은 몇 가지 중요한 시사점을 제공한다.

첫째, 특정 보안 통제항목에 CVE가 집중되는 현상은 해당 통제항목이 더 많은 취약점을 포함할 수 있다는 것을 의미하며, 해당 통제항목이 실제로 중요한 보안 요소이므로 보안 관리에서 우선적으로 강화해야 할 필요성을 시사한다.

둘째, CVE 분포의 불균형은 NIST SP 800-53이 최신 취약점을 완전히 반영하지 못했거나, 특정 취약점이 보안 통제항목에서 충분히 다루어지지 않았을 가능성을 제기한다. 이는 NIST SP 800-53의 설계가 최신 보안 위협을 충분히 고려하지 않았을 수 있음을 암시하며, 보안 표준의 업데이트 필요성을 제기할 수 있다.

셋째, 본 연구의 범위가 우분투 운영체제에 한정되어 있어 다른 산업 분야나 운영체제에 대한 일반화에 한계가 있다. 이는 연구 결과의 외부 타당성을 높이기 위해 다양한 산업 데이터와 다른 운영체제에 관한 추가 연구가 필요함을 시사한다.

따라서 향후 연구에서는 본 연구의 한계를 보완하기 위해 다음과 같은 방향으로 진행할 수 있다. 첫째, NIST SP 800-53의 보안 통제항목이 최신 CVE를 얼마나 효과적으로 반영하고 있는지에 대한 심층 분석이 필요하다. 둘째, 다양한 산업 분야와 운영체제에 대해 보안 강도 네트워크를 확장하여 분석함으로써 연구 결과의 외부 타당성을 높여야 한다. 셋째, 보안 통제항목과 CVE 간의 관계를 보다 정교하게 모델링 하여, 보안 통제의 우선순위를 설정하고 효과적인 보안 관리 방안을 제시할 필요가 있다. 향후 확장된 연구 방향을 통해 보안 강도 평가의 신뢰성과 적용 범위를 확장하고, 조직의 보안 상태를 체계적으로 관리할 수 있을 것으로 기대된다.

V. Conclusion

본 논문에서는 NIST SP 800-53과 CVE 간의 상호 연관성을 분석하여 보안 통제의 우선순위를 설정하여 중요한 평가항목을 도출하였다.

CVE와 NIST SP 800-53과의 관계를 도출하기 위하여, STIG와 OVAL 간의 매핑을 보안에 특화된 인지능 모델 SecBERT, CyBERT, RankT5를 사용하였다. 그 결과 제로샷 설정을 한 RankT5, 파인튜닝된 SecBERT(F1 점수 92.50%), CyBERT(F1 점수 79.65%) 순으로 정확도가 높았다. RankT5 모델은 파인튜닝을 하지 않았음에도 모든 지표에서 100%의 성능을 보였으나, 과적합 가능성이 있어 현실적인 성능을 보인 SecBERT 모델을 최종 결과로 채택하였다.

본 연구는 정보 시스템 보안 평가의 정확성과 완전성을 높이고, 보안 관리자들이 취약점을 효과적으로 관리하여 NIST SP 800-53의 중요한 통제 항목을 적용함으로써 정보 시스템의 보안 수준을 향상시키는 데 기여할 것으로 기대된다.

ACKNOWLEDGEMENT

This work was supported by the research grant of the Kongju National University in 2023.

REFERENCES

[1] CVE Details. “Vulnerabilities By Types/Categories”. https://www.cvedetails.com/vulnerabilities-by-types.php.

[2] National Institute of Standards and Technology (NIST). Security and Privacy Controls for Information Systems and Organizations: NIST Special Publication 800-53, Revision 5. Gaithersburg, MD: National Institute of Standards and Technology, September 2020. DOI: 10.6028/NIST.SP.800-53r5.

[3] H. Na and H. Jung. "A Theoretical Comparative Study of Human Resource Security Based on Korean and Int'l Information Security Management Systems." Journal of Convergence for Information Technology, Vol. 6, No. 3, pp. 13?19, September 2016. DOI: 10.22156/CS4SMB.2016.6.3.013.

[4] S. Kim. “A Comparative Study on Information Security Management Activity of Public Sector in USA & Korea." The KIPS Transactions: Part C, Vol. 13C, No. 1, pp. 69?74, February

2006. DOI: 10.3745/KIPSTC.2006.13C.1.069.

[5] “Control Correlation Identifier(CCI) Process,” version 1 release 0.1, pp. 1-5, February 2011. https://dl.dod.cyber.mil/wp-content/uploads/stigs/pdf/u_cci_process_v1r0.1.pdf.

[6] "Control Correlation Identifier," DoD Cyber Exchange Public. https://public.cyber.mil/stigs/cci/.\

[7] National Institute of Standards and Technology. "Control Correlation Identifier (CCI)." NIST Computer Security Resource Center. https://csrc.nist.gov/glossary/term/CCI.

[8] Security Requirements Guide. NIST Computer Security Resource Center. https://csrc.nist.gov/glossary/term/security_requirements_guide.

[9] Efense Information Systems Agency. "Security Technical Implementation Guides (STIGs)." DoD Cyber Exchange. https://public.cyber.mil/stigs/.

[10] MITRE Corporation. "CVE Numbering Authorities (CNAs)." CVE Program. https://cve.mitre.org/cve/cna.html.

[11] ational Institute of Standards and Technology. OVAL Language Specification, Version 5.11.3. MITRE Corporation, 2020. https://oval.mitre.org/language/about/specification.html.

[12] Hamdani, S. W. "Framework for Assessing Information System Security Posture Risks." Master's thesis, The University of Western Ontario, June 2023.

[13] Branescu, I., Grigorescu, O., and Dascalu, M. "Automated Mapping of Common Vulnerabilities and Exposures to MITRE ATT&CK Tactics." Information, Vol. 15, No. 4, pp. 214, 2024. DOI: 10.3390/info15040214.

[14] Haddad, A., Aaraj, N., Nakov, P., and Mare, S. F. "Automated Mapping of CVE Vulnerability Records to MITRE CWE Weaknesses." arXiv, April 2023. https://arxiv.org/abs/2304.11130.

[15] Red Hat, Inc. "OVAL Repository." https://access.redhat.com/security/data/oval/.

[16] Canonical Ltd. "Ubuntu OVAL Data." Ubuntu Security. https://ubuntu.com/security/oval.

[17] National Institute of Standards and Technology. "National Checklist Program Repository." NIST. https://nvd.nist.gov/ncp/repository.

[18] Cui, Y, Jia, M.l Lin, T, Song, Y. and Belongie, S. "Class-Balanced Loss Based on Effective Number of Samples." arXiv preprint arXiv:1901.05555, 2019. https://ar5iv.labs.arxiv.org/html/1901.05555.

Authors

Se-Eun Kim received the B.S. in Information and Communication Engineering and M.S. in Artificial Intelligence from Kongju National University, Korea in 2018 and 2024, respectively.

Her main research interests include embedded systems, computer systems, and industrial control system security.

Hyo-Beom Ahn received the B.S. in Computer Science and M.S., and Ph.D. in Computer Science and Statistics from Dankook University, Korea in 1992, 1994 and 2002 respectively.

Since then, he has been with the Divsion of Artificial Intelligence Kongju National University Rep. of KOREA. His main research interests include Computer Networks, Network Security Smart Grid Security and Application and Industrial control system security.