국가용 보안요구사항 V3.0 기반의

국내용 취약성 분석서 작성 가이드 V1.0

2025. 4. 11.

과학기술정보통신부

kisia 한국정보보호산업협회

[문서 변경이력]

버전	일 자	변경 내용
1.0	2025. 4. 11.	o v1.0 문서 배포

목 차

1. 개요

1.1. 취약성 분석서 참조

☞ *취약성 분석서 식별을 명확하게 할 수 있도록 문서명, 문서버전 등 취약성 분석서에 대한 식별정보 서술*

문서명	*OOO 취약성 분석서*
문서버전	*X.X*
파일명	*신청대상제품명-문서명-버전-날짜(세부버전 등).hwp*
보증컴포넌트	*AVA_VAN.2*
작성자	*(주) XXX*
발간일	*20XX.XX.XX*
주요용어	*침입차단시스템, 방화벽, FW, Firewall 등*

참고사항	▶ 정보보호제품 평가인증(CC인증) 제도의 국내용 정보보호제품 평가·인증 스킴에 따라 개발자 취약성 분석서는 보안구조설명 등 개발문서 제출로 인정된다. ▶ 취약성 분석서는 작성기관의 자체 양식을 이용하여 작성될 수 있으며, 보안 구조 설명에 대한 내용 및 알려진 취약점 등 TOE에 대한 취약점 분석 내용을 포함해야 한다. ▶ 신규로 제출물을 작성하는 경우 취약성 분석서 작성자는 본 가이드를 참고하여 취약성 분석서를 작성할 수 있다.

1.2. 목적

☞ *취약성 분석서는 자체보호, 영역분리, 우회불가성 등 TSF의 보안 구조에 대한 설명을 제공하고 TOE 보안구조, 알려진 취약점 등에 대한 취약점을 식별 및 분석 결과를 제공하여, 시험 및 취약성 분석 평가를 위한 기초 자료로 사용된다.*

☞ *자체보호, 영역분리, 우회불가성 등 TSF가 손상되거나 우회되지 않는 방법 등 보안 구조에 대한 설명을 제공하고, 각 특성들을 TSF 초기화 과정이 안전하게 제공됨을 설명한다.*

예 : 본 문서는 [OOO V1.0]의 보안 아키텍처를 설명하고, TSF가 자체 보호, 영역분리, 우회불가성을 제공하기 위한 정보를 제공합니다.
본 문서는 EAL2 평가를 위한 ADV_ARC.1 요구사항을 충족하기 위해 작성되었습니다.

1.3. 취약점 분석서 구성

취약점 분석서는 다음과 같이 구성되어 있다.

1장은 취약성 분석서 개요로 취약성 분석서의 목적, 구성, 용어 정의 등을 기술한다.

2장은 TOE 보안 구조에 대한 설명을 기술하고, 보안 구조에 기반한 취약점 분석 내역을 기술한다.

3장은 TOE에 해당되는 공개된 취약점에 대한 분석 내역을 기술한다.

1.4. 용어 정의

☞ *취약성 분석서에서 사용되는 용어를 중점적으로 추가하여 기술한다.*

예:

우회불가성(non-bypassability)

SFR과 관련된 모든 행동은 TSF를 통해 이루어져야 한다는 보안구조 특성

TSF 자체보호(TSF self-protection)

TSF는 비-TSF 코드 또는 실체에 의해서는 손상될 수 없다는 보안 구조 특성

취약성(vulnerability)

어떤 환경에서 SFR을 위반하는 데 사용될 수 있는 TOE의 약점

악용 가능한 취약성(exploitable vulnerability)

TOE의 운영환경에서 SFR을 위반하는 데 사용될 수 있는 TOE의 약점

잠재적 취약성(potential vulnerability)

확인되지는 않았으나 (가정된 공격 경로를 통해) SFR을 위반할 것으로 의심되는 약점.

잔여 취약성(residual vulnerability)

TOE의 운영환경에서 악용될 수 없지만, TOE의 운영환경에서 예상되는 것보다 더 높은 수준의 공격 성공 가능성을 가진 공격자가 SFR을 위반하는 데 사용할 수 있는 약점

1.5. 참조 문서

☞ *일반적으로 보안 구조 설명(ADV_ARC.1)에서 요구되는 입력문서를 참조로 한다.*

- OOO V1.0" 보안목표명세서 VX.X
- OOO V1.0" 기능명세서 VX.X
- OOO V1.0" TOE 설계서 VX.X
- OOO V1.0" 사용자 운영 설명서 VX.X

2. 보안 구조 설명

본 장에서는 TOE의 보안 구조에 대한 설명을 기술한다. 보안 구조는 자체보호, 영역분리, 우회불가성을 포함한 TSF 특성이 포함되며, 각 특성이 제공됨을 입증하기 위한 설명을 기술한다.

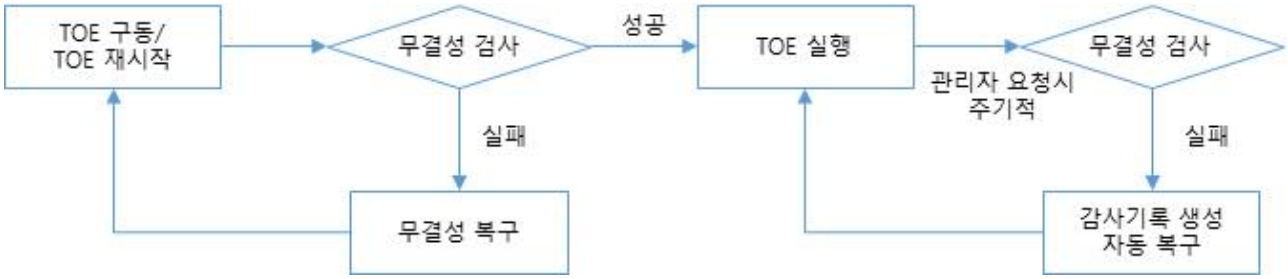
☞ 보안 구조 설명은 공통평가기준 3부 “부록 A.1.2 보안 구조 설명”에 따라 자체 보호, 영역분리, 우회불가성에 대한 특성을 시험할 수 있을만큼 충분히 상세할 것을 요구한다.

2.1. 자체보호

☞ 자체보호란 TSF에 변경을 일으킬 수 있는 외부 실체의 조작으로부터 자신을 보호하는 TSF의 능력을 의미한다.

예 : TOE는 무결성 검사 기능을 통해 자체 보호 기능을 제공한다.

- 무결성 검사 프로세스 : TOE는 구동시 자체 무결성을 검사를 수행하며, TOE의 무결성이 손상된 경우 자동복구 기능을 제공한다.



1) TOE는 구동시 TSF 실행파일 및 환경설정에 대한 무결성 검사를 수행하고, 무결성 검사 실패시 무결성 복구를 수행한 후 TOE를 재구동한다.

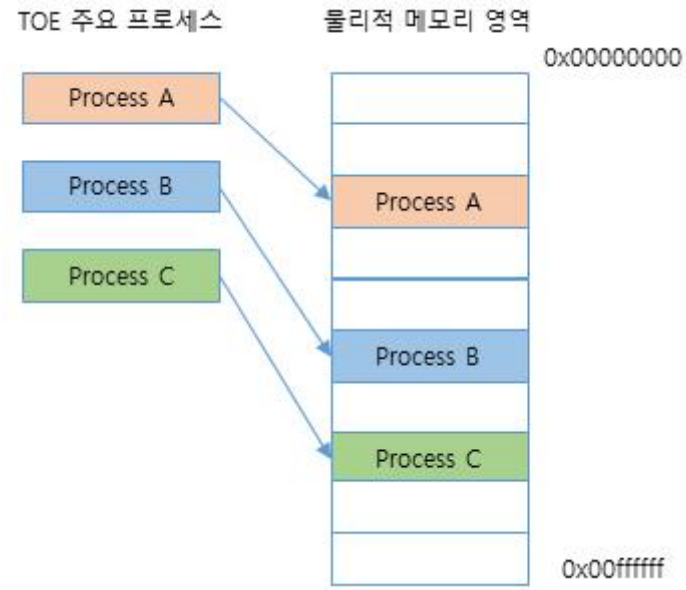
2) TOE는 관리자 요청시 또는 주기적으로 무결성 검사를 수행하며, 무결성 손상이 탐지된 경우 감사기록을 생성하고,자동 복구를 수행한다.

2.2. 영역분리

☞ 영역분리는 각 신뢰되지 않은 실체에 대해 TSF 자원을 운영하기 위한 분리된 보안영역을 생성하는 특성으로, 이러한 영역은 다른 영역과 분리되며 어떠한 실체도 다른 영역에 접근할 수 없도록 유지된다. 예를 들어, 운영체제인 TOE는 신뢰되지 않은 실체와 연관된 프로세스에 대한 영역(주소 공간, 프로세스별 환경 변수)을 제공한다

예 : TOE는 주소 공간 분리를 통해 영역 분리를 제공한다.

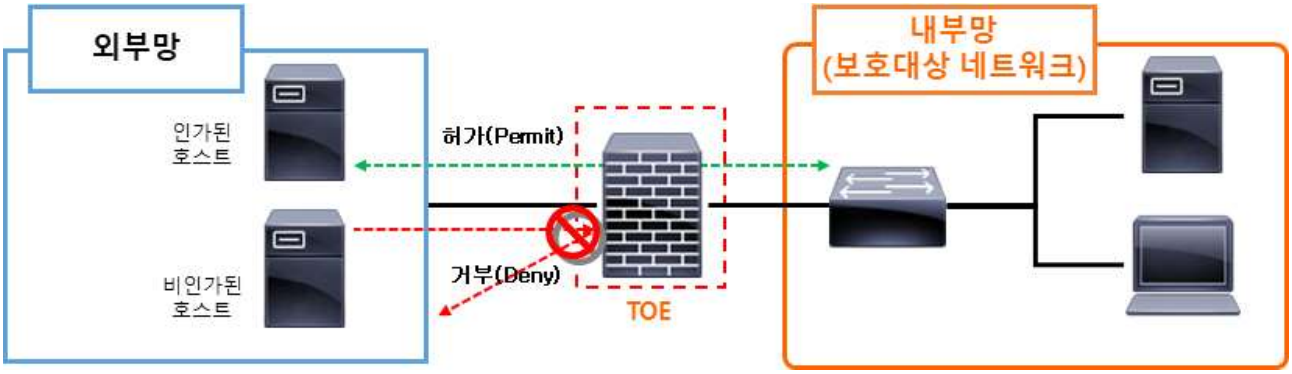
- 주소 공간 분리 : TOE는 주소 공간 관리를 통해 실행중인 프로세스가 다른 프로세스의 메모리 주소 공간에 접근을 통제하여, 프로세스간 상호 간섭을 방지한다.



2.3. 우회불가성

☞ 우회불가성은 특정 메커니즘이 적용되는 경우 TSF의 보안기능성(SFR에서 명시된 대로)이 항상 호출되어 우회될 수 없는 특성이다.
예를 들어, 파일에 대한 접근통제가 SFR에 따라 TSF의 기능으로 명시된 경우, TSF의 접근통제 메커니즘을 호출하지 않고 파일에 접근할 수 있는 인터페이스는 존재하지 않아야 한다.

예 : TOE는 침입차단제품으로 NIC 통해 외부네트워크에서 유입되는 모든 패킷은 OS 커널 영역에서 수행되는 보안정책을 통해서만 유입되도록 하여, 보안정책에 대한 우회불가성을 제공한다.



2.4. 보안 구조 설명 기반 취약점 분석

☞ 보안 구조 설명에 기반한 취약점을 식별 및 분류하고, 이에 대한 분석 결과를 서술한다.

☞ 자체보호, 영역분리, 우회불가에 대한 취약점 점검 항목, 점검 내용, 점검 결과를 서술한다.

예) 자체보호, 영역분리, 우회불가성 등 보안 구조 설명에 기반한 취약점 분석 결과는 다음의 표와 같다.

구분	취약점 점검 항목	점검 내용	점검 결과
자체보호	TOE 설정 파일 위변조	TOE의 주요 실행파일 및 환경 설정 파일을 임의로 변조한다.	환경 설정 파일에 대한 쓰기 방지 및 주기적인 무결성 검사를 수행하므로 취약하지 않음
영역분리	버퍼오버플로우	TOE 주요 프로세스에서 사용되는 함수에 버퍼 크기를 초과하는 입력값을 설정하여 오버플로우 여부를 확인한다.다.	버퍼사용시 입력값에 대한 검증을 수행하므로, 버퍼오버플로우에 취약하지 않음
우회불가	방화벽 정책 우회	TOE 부팅/재부팅 시 방화벽 정책을 우회하여 네트워크 통신 가능여부를 확인한다.	TOE 부팅/재부팅 시 NIC를 통한 네트워크 통신이 제한되므로 취약하지 않음

2.4.1 TOE 설정 파일 위변조

점검 내용
☞ 취약점 점검 항목에 대한 점검 내용을 기술한다. TOE의 주요 실행파일 및 환경 설정 파일을 임의로 변조하여 자체보호 기능을 확인한다. - TOE 주요 실행파일 : /TOE/Bin/Process_A - 환경설정 파일 : /TOE/Conf/config.cfg
점검 결과
☞ 취약점 점검을 수행된 각 단계별 절차 및 점검결과를 기술한다. 1. TOE의 주요 실행파일 및 환경 설정 파일을 확인한다. 2. 관리자 계정으로 TOE 주요 실행파일 및 환경 설정 파일에 임의로 변조한다. - 주요 실행파일 및 환경설정 파일에 대한 쓰기 방지로 변조에 실패한다. 3. root 계정으로 TOE 주요 실행파일 및 환경 설정 파일에 임의로 변조한다. - 실행중인 주요 실행파일에 대한 쓰기 방지로 변조에 실패한다. - 변조된 환경 설정파일에 대한 무결성 검사 결과가 관리자에게 통보되고, 감사기록이 생성된다.

3. 취약점 분석

본 장에서는 알려진 취약점, 제품 유형, 자체 취약점 DB 등 TOE에 대한 취약점을 분석하고, 이에 대한 개선결과를 기술한다. 식별된 취약점이 TOE에 영향이 없을 경우, 관련근거를 함께 작성한다.

3.1. 취약점 목록

☞ *CVE, 개발사, KISA 홈페이지 등 알려진 취약점 중 TOE에 해당되는 취약점을 식별하고 이에 대한 개선 내역을 작성한다.*

구분	식별자	대상	제품구성요소	위협정도	분석결과
1	CVE-2023-5678	OpenSSL	A v1.0.0	중	취약하지 않음/개선완료 분석결과 : 3.2.1 참조
2	CVE-2022-0480	Linux Kernel		중	
3	CVE-2021-44228	Log4j	B v1.0.1	상	
4	CVE-2019-12984	Linux Kernel	C v1.1.0	중	
5	파일 업로드	-	A v1.0.0	중	

3.2. 취약점 분석 결과

☞ *식별된 취약점에 대한 분석 결과를 작성한다.*

3.2.1. OpenSSL 취약성(CVE 2023-4807)

식별자	CVE-2023-5678	위협 정도	중
출처	☞ *식별된 취약점에 대한 출처를 작성한다.* https://nvd.nist.gov/vuln/detail/CVE-2023-5678		
설명	☞ *취약점에 의해 영향을 받는 기능을 기재한다.* *예) 식별 및 인증, 암호화 기능 등* ☞ *CVE 등재 취약점의 경우 영어로 기술 가능* 비정상적으로 긴 X9.42 DH 키 생성시 처리시간이 지연이 발생하는 취약점으로, 서비스 거부(DoS) 공격이 발생 될 수 있음 Generating excessively long X9.42 DH keys or checking excessively long X9.42 DH keys or parameters may be very slow. Impact summary: Applications that use the functions DH_generate_key() to generate an X9.42 DH key may experience long delays. Likewise, applications that use DH_check_pub_key(), DH_check_pub_key_ex() or EVP_PKEY_public_check() to check an X9.42 DH key or X9.42 DH parameters may experience long delays. Where the key or parameters that are being checked have been obtained from an untrusted source this may lead to a Denial of Service. While DH_check() performs all the necessary checks (as of CVE-2023-3817), DH_check_pub_key() doesn't make any of these checks, and is therefore vulnerable for excessively large P and Q parameters. Likewise, while DH_generate_key() performs a check for an excessively large P, it doesn't check for an excessively large Q. An application that calls DH_generate_key() or DH_check_pub_key() and supplies a key or parameters obtained from an untrusted source could be vulnerable to a Denial of Service attack. DH_generate_key() and DH_check_pub_key() are also called by a number of other OpenSSL functions. An application calling any of those other functions may similarly be affected. The other functions affected by this are DH_check_pub_key_ex(), EVP_PKEY_public_check(), and EVP_PKEY_generate(). Also vulnerable are the OpenSSL pkey command line application when using the "-pubcheck" option, as well as the OpenSSL genpkey command line application. The OpenSSL SSL/TLS implementation is not affected by this issue. The OpenSSL 3.0 and 3.1 FIPS providers are not affected by this issue.		
개선 내용	☞ *취약점 에 대한 개선 내용을 기재한다.* [취약점에 대한 개선이 발생되지 않은 경우] 해당 취약점은 OpenSSL 3.0.1.에서 발생하는 취약점으로 TOE는 해당 취약점이 제거된 OpenSSL 3.0.13 버전을 사용하므로 해당사항 없음		

	[취약점에 대한 개선이 발생된 경우] crypto/dh/dh_check.c 및 crypto/dh/dh_key.c 파일에서 취약점 발생되며, 비정상적으로 긴 X9.42 DH 키 및 파라미터가 입력될 경우, Error 처리하도록 수정 [수정내역]