

2021 KISA REPORT

volume 12

www.kisa.or.kr

CONTENTS

ISSUE I. 디지털

- 01 차기 정부의 IT 정책에 바라는 것
[한상기/ 테크프런티어 대표]
- 02 빅테크 기업 규제의 방향성
[최호섭/ IT칼럼니스트]
- 03 무한경쟁 OTT 시장 라운드1 전망
[최홍규/ EBS 연구위원]

ISSUE II. 정보보호

- 04 소프트웨어 보안 관점에서 본 미국 사이버보안 행정명령과 우리의 대응방안
[이태준/ 고려대학교 컴퓨터보안연구실 연구원]
- 05 정보시스템의 상시 모니터링을 위한 SCAP과 도구에 대한 소개
[안효범/ 공주대학교 인공지능학부 교수]

ISSUE III. 개인정보보호

- 06 개인정보 정책에서 증거기반(evidence-based)의 규제의 필요성
[이진규/ 네이버주식회사 이사]

주제 제안 및 정기 메일 신청 | kisareport@kisa.or.kr

인터넷 정보보호 관련 이슈, 현안 등 궁금한 내용을 보내주시면 선별 후 보고서 주제로 선정됩니다.

또한, KISA Report 온라인 서비스 제공을 원하실 경우 신청해주시면 매월 받아보실 수 있습니다.

정보시스템의 상시 모니터링을 위한 SCAP과 도구에 대한 소개



안효범 (hbahn@kongju.ac.kr)

공주대학교 인공지능학부 교수

정보시스템에 대한 상시 모니터링을 통해 보안체계를 유지하는 것은 중요한 정보보호 활동 중에 하나의 작업이다. 상시 모니터링을 통해, 정보시스템에 대한 취약점을 발견할 수 있고 빠르게 대처할 수 있는 정보보호체계를 구성할 수 있게 된다. 그러나, 대부분의 기업들은 침입을 방어하는 입장에서 정보보호체계를 유지하려는 특성을 가진다. 즉, 사고가 발생했을 때, 정보보호를 위한 대응체계를 가동하게 된다. 적극적인 정보보호 활동을 수행하기 위해서는 상시 모니터링을 할 수 있는 표준이 필요하며, 이를 위한 도구가 필요한데, 이 논고에서는 상시모니터링을 위한 표준으로 제시되고 있는 SCAP(Security Content Automation Protocol)에 대한 구성을 알아보고, 검증에 사용될 수 있는 도구들에 대하여 소개한다.

1. 보안 표준화 및 자동화 동향

보안 자동화(security automation)는 특정 공동 보안 기능을 수행하기 위해 표준화된 규격 및 프로토

콜을 사용하는 것으로 취약성 평가 소프트웨어를 예로 들 수 있다. 취약성 평가 소프트웨어는 각 정보시스템에 대해 검사를 수행하여 패치되지 않거나 잘못된 보안 구성 설정(운영체제와 설치된 소프트웨어의 환경설정파일)여부와 같은 일련의 취약성을 확인한다. 보안 자동화 기술은 자산 관리(asset management), 구성 관리(configuration management), 취약점 관리(vulnerability management)가 상호 작용하면서 시스템을 효과적으로 관리하고 모니터링 할 수 있다.

각 보안 자동화 기술 구성요소는 개별적으로 사용할 때 유용하지만 취약점을 검증과 보고를 하기 위한 여러 표준을 함께 사용하면 큰 이점을 갖게 된다. 많은 기업과 벤더는 이러한 표준을 결합하고 구성요소가 함께 작동할 수 있도록 하는 방법을 보안 자동화 프로토콜(security automation protocol)로 정의해 왔다. 하나의 표준은 여러 프로토콜에서 사용되며 각 프로토콜은 특정 목적을 위해 구성요소들이 함께 사용되는 방법을 정의함으로써 구성요소 표준을 기반으로 한다. 보안 자동화 프로토콜 중 가장 잘 알려진 것은 SCAP이다. SCAP은 현재 버전이 1.3까지 NIST SP-800-125 Revision3에서 정의하고 있다.

2. SCAP의 구성 및 구조

SCAP은 컴포넌트 명세(component specification)들의 다중-목적 프레임워크이다. 컴포넌트 명세는 자동화된 환경설정, 취약점 그리고 패치 검사, 보안성 측정(security measurement) 그리고 기술적 제어를 위한 내부통제(compliance) 활동들을 지원한다¹⁾. SCAP 1.3은 5가지의 범주에 대하여 20개의 컴포넌트로 구성되어 있다. [표 1]은 SCAP의 5가지의 범주에 대하여 요약을 하였다.

언어 범주에서는 SCAP에서 사용되는 표준 어휘들과 보안 정책을 표현하기 위한 규칙들, 기술적 검사 메카니즘 그리고 수행된 결과들을 제공한다. 보고 형식 같은 경우에는 표준화된 형식으로 수집된 정보를 표현하기 위한 필수적인 구조를 제공한다. 식별 체계에서는 표준 식별 형식을 사용하여 소프트웨어 제품, 취약점 그리고 환경설정 아이템들 같은 주요 개념들을 식별하기 위한 방법들을 제공한다. 측정과 스코어링 시스템에서는 보안 약점(예를 들면, 소프트웨어 취약점과 보안 환경설정 이슈들) 중의 특정 특성들을 평가하기 위하여 참조된다. 무결성에서는 SCAP의 내용들과 결과의 무결성을 보존하기 위하여 사용된다.

1) D. Waltermire, S. Quinn, H. Booth, K. Scarfone, and D. Prisaca, "The Technical Specification for the Security Content Automation Protocol (SCAP): SCAP Version 1.3," NIST Special Publication 800-126. Revision 3, National Institute of Standards and Technology, February 2018.

[표 1] SCAP 버전 1.3의 5가지의 범주와 포함된 구성요소

카테고리	구성요소	설명
언어 (language)	XCCDF (Extensible Configuration Checklist Description Format)	보안 체크리스트/벤치마크 작성 및 평가 결과 보고용 언어로, 보안 검증 항목에 관한 내용을 작성하는 데 사용되며 검증 이후 해당 검증 결과를 표현하는 언어
	OVAL (Open Vulnerability and Assessment Language)	시스템 구성 정보 표현, 시스템 상태 평가 및 평가 결과 보고를 위한 언어로, XML 형식으로 구성
	OCIL (Open Checklist Interactive Language)	다른 데이터 수집 노력으로 만들어진 기존의 데이터 저장소 또는 사용자로부터 정보를 수집한 데이터를 나타내는 언어
보고 형식 (reporting format)	ARF (Asset Reporting Format)	자산(asset)에 대한 정보의 전송 형식 및 자산과 보고서 간의 관계를 표현하기 위한 형식
	AI (Asset Identification)	자산(asset)에 대한 알려진 식별자 및 알려진 정보를 기반으로 자산을 고유하게 식별하기 위한 형식
식별 체계 (identification scheme)	CPE (Common Platform Enumeration)	하드웨어, 운영 체제, 애플리케이션의 명명 및 사전으로, SCAP에서는 해당 플랫폼이 무엇인지에 대한 명시를 위해 CPE 이용
	CCE (Common Configuration Enumeration)	보안 관련 소프트웨어 결함의 명명 및 사전으로, SCAP에서는 보안 관련 시스템 설정 항목을 확인하기 위해 CCE 이용
	CVE (Common Vulnerabilities and Exposures)	소프트웨어 보안 구성의 명명 및 사전으로, 제품의 취약점에 대해 고유 식별 번호 'CVE 식별 번호'를 부여함으로써 정보 간의 상호 참조 및 연결에 CVE 사용
	SWID (Software Identification)	소프트웨어 식별자 및 관련 메타데이터를 나타내기 위한 형식
측정 및 채점 시스템 (measurement and scoring system)	CVSS (Common Vulnerability Scoring System)	소프트웨어 결함 취약성의 상대적 심각도를 측정하기 위한 시스템
	CCSS (Common Configuration Scoring System)	시스템 보안 구성 문제의 상대적 심각도를 측정하기 위한 시스템
무결성 (integrity)	TMSAD (Trust Model for Security Automation Data)	다른 보안 자동화 규격에 적용되는 공통 신뢰 모델에서 디지털 서명을 사용하기 위한 규격

SCAP은 보안취약점에 대한 점검을 수행하고 평가하기 위한 일관된 방법을 제공하기 위해

열거(enumeration), 언어(language), 위험측정(risk measurement)의 세 가지 종류의 자동화 표준을 사용한다²⁾.

가) 열거 표준 (Enumeration Standards)

보안 열거(enumeration) 표준은 보안 자동화 요소에 대한 명명법이며 해당 명명법을 사용하여 표현된 항목의 사전도 제공한다. 가장 널리 사용되는 보안 열거 표준은 CVE(Common Vulnerabilities and Exposures)로, 공개적으로 발표된 소프트웨어 취약성에 대한 고유 식별자를 제공한다. CVE는 식별자는 취약점 스캐너 및 침입 탐지 시스템 등에서 취약점을 참조할 때 사용되며 소프트웨어 판매 회사들은 제품의 새로운 취약점에 대한 권고 사항을 발행할 때 CVE 식별자를 포함한다. CVE와 유사한 열거 표준으로 CCE(Common Configuration Enumeration)와 CPE(Common Platform Enumeration)가 있다. CCE는 특정 운영체제의 개별 구성 설정과 같은 소프트웨어 보안 구성 문제에 대한 고유 식별자를 제공하며, CPE는 하드웨어, 운영체제 및 응용 프로그램 버전에 대한 고유 식별자를 제공한다.

나) 언어 표준 (Language Standards)

보안 언어 표준은 보안 정책, 보안 검사 목록(checklist), 개별 보안 항목에 대한 검사를 수행하는 메커니즘과 같은 보안 정보를 표현하기 위해 표준화된 어휘 및 규칙을 제공한다. 앞서 소개한 열거 표준을 통해 검사할 항목을 지정할 수 있는 경우, 언어는 일련의 검사를 프로파일에 제시하거나 해당 검사를 수행하는 데 사용된 기술적 메커니즘에 대해 제품에 규정함으로써 검사 방법을 자세히 설명한다.

(1) 체크리스트 언어 표준

체크리스트 언어 표준은 체크리스트 작성 및 실행에 사용되며 대표적으로 XCCDF(Extensible Configuration Checklist Description Format)가 있다. XCCDF는 가장 많이 사용되는 보안 체크리스트 언어지만 보안 목적이 아닌 체크리스트 작성 및 실행에도 사용될 수 있을 만큼 유연하다.

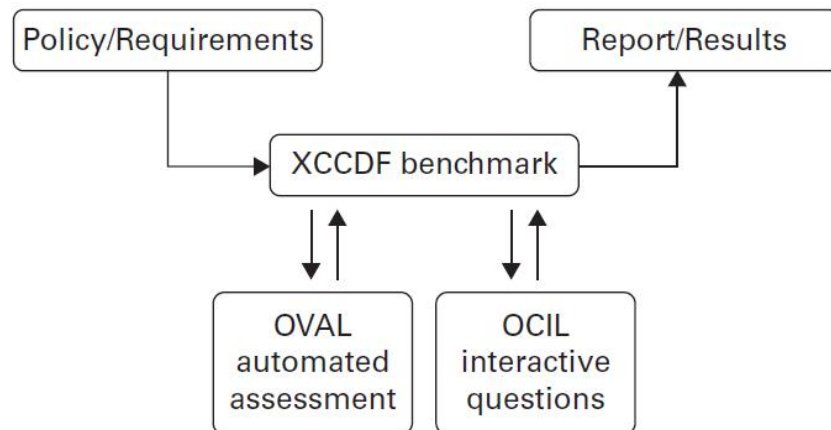
(2) 검사 언어 표준

검사를 위한 언어 표준은 보안 설정 확인, 알려진 취약점 검색, 패치 여부 확인, 컴퓨터에서 기타 정보 수집과 같은 보안 검사를 수행하기 위해 설계되었으며 대표적으로 OVAL(Open Vulnerability and Assessment Language)이 있다. OVAL은 개별 보안 검사와 검사 결과 보고 시 사용된다. 개별 OVAL 검사를 직접 평가할 수도 있지만, 전체 검사 목록을 단일 XCCDF 체크 리스트로 컴파일하고 XCCDF가 필요한 OVAL 테스트를 호출하도록 하는 것도 가능하다.

OCIL(Open Checklist Interactive Language)은 OVAL을 보완하기 위해 추가로 만들어진 보안 언어 표준이다. 다른 자료수집 작업을 통해 사용자나 기존 데이터 저장소로부터 정보를 수집하는 검사를 나타

2) S. Radack, "Security Content Automation Protocol (SCAP): Helping Organizations Maintain and Verify the Security Of Their Information Systems", ITL Bulletin, September 2010.

내는 언어로 설문 형식의 보안 평가를 수행할 수 있는 언어다. 예를 들어 OCIL을 사용하여 사용자에게 보안 관련 질문을 할 수 있다. OVAL과 마찬가지로 OCIL 설문지도 개별적으로 호출하거나 XCCDF 체크 리스트로 호출할 수 있다. 단일 XCCDF 체크 리스트는 OCIL 설문지와 OVAL 검사 및 기타 검사 언어 표준을 모두 호출할 수 있으며, 모든 검사 결과를 포함하는 단일 보고서를 작성할 수 있다. [그림 1]과 같이 XCCDF는 OVAL 및 OCIL과 연계하여 필요한 검사를 수행하고 검사 결과를 반환한다.



[그림 1] XCCDF, OVAL, OCIL의 상호 작용

(3) 이벤트 언어 표준

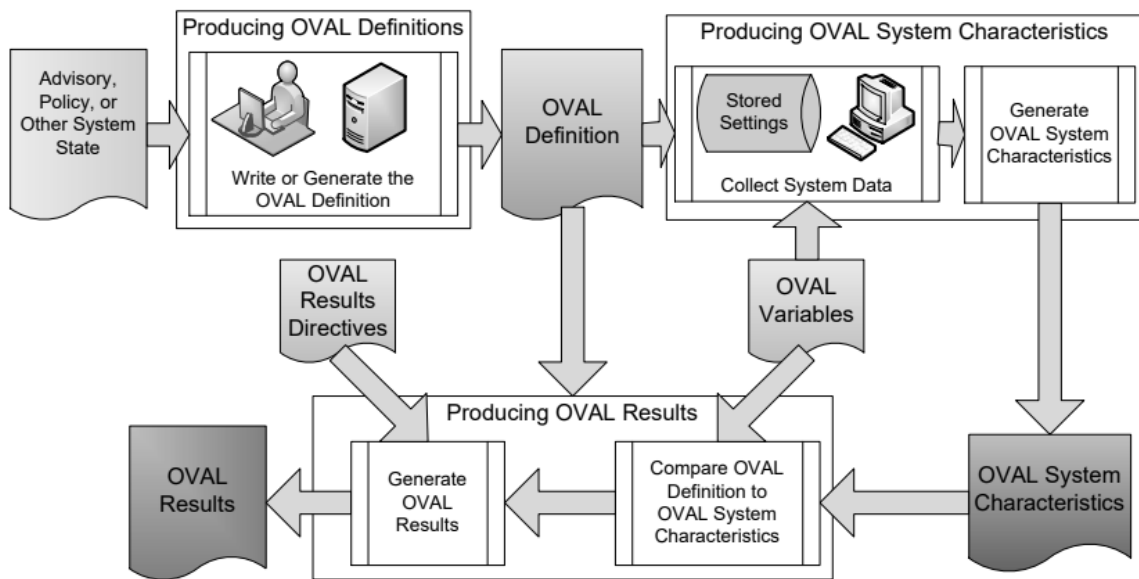
이벤트 언어 표준은 보안 이벤트를 포함하여 컴퓨터 이벤트의 특성을 문서화하는 데 사용되며, 특히 로그 파일에서 표준화된 형식으로 표현될 수 있다. 관련 이벤트 언어 표준으로 이벤트 필드 사전에 포함하는 CEE(Common Event Expression)가 있다. CEE는 주목할 만한 이벤트가 발생했다고 보고하는 공통 언어를 제공하고 이벤트를 알아야 하는 사람들, 공유하는 방법을 정의하고, 수신 시스템이 이벤트를 해석하는 방법을 설명한다. 이를 통해 보안 자동화 커뮤니티가 효과적으로 정보를 공유할 수 있도록 돕는다.

(4) 자산 언어 표준

자산 언어 표준은 컴퓨터, 네트워크, 소프트웨어 및 하드웨어를 포함한 다양한 자산(asset)과 관련된 정보를 문서화하기 위한 프레임워크를 제공한다. 자산 식별(Asset Identification, AI) 표준을 사용하면 표준화된 표현을 사용하여 자산을 고유하게 식별할 수 있다. AI의 보완적 표준인 자산 보고 형식(Asset Reporting Format, ARF)은 표준화된 보고 형식과 자산에 대한 정보를 한 컴퓨터에서 다른 컴퓨터로 전송할 수 있는 방식으로 표현하는 방법을 정의한다. 이를 통해 보안 정보를 포함한 자산에 대한 정보를 보고할 수 있다.

3. 보안 평가를 위한 OVAL에 대하여

OVAL(Open Vulnerability and Assessment Language)은 컴퓨터 시스템의 상태를 평가하고, 취약점을 보고하는 과정을 표준화하기 위하여 생성된 국제 정보보안 커뮤니티 표준으로, 시스템 관리자가 로컬 시스템상에서 시스템이 가지고 있는 특성 및 설정 정보들을 기반으로 취약점을 찾아내고, 평가하는 과정을 거칠 수 있도록 한다. OVAL 언어는 시스템의 설정 정보를 수집하여 보안 정책을 준수하는지 테스트한 뒤 결과를 XCCDF 컴포넌트로 넘기는 등 실질적으로 시스템을 평가하는 과정에 쓰이는 언어이며, 그러한 점에서 SCAP 컴포넌트들 중 가장 중요한 역할을 하는 컴포넌트로 결론 지을 수 있다.



[그림 2] OVAL 언어 작동 과정

OVAL을 이용하여 시스템을 평가하는 프로세스는 그림 36에서 나타나듯 크게 세 단계로 구분된다. 평가하고자 하는 시스템의 설정 파일과 경로를 확인한 뒤, 수집하고자 하는 객체와 해당 객체의 상태를 OVAL 코드로 작성한다. 보안 스캐너 내에서 작성된 OVAL 코드를 이용하여 객체를 수집하고, 수집된 객체의 존재 여부나 상태를 이용하여 평가를 수행한다. 평가된 결과는 보안 스캐너에 의해 기본적으로 OVAL Results라는 XML 파일로 작성되며, 해당 결과를 XCCDF 컴포넌트에 넘겨 테스트 결과를 최종적으로 보고하여 시각적으로 리포트를 표현하는 과정을 거친다.

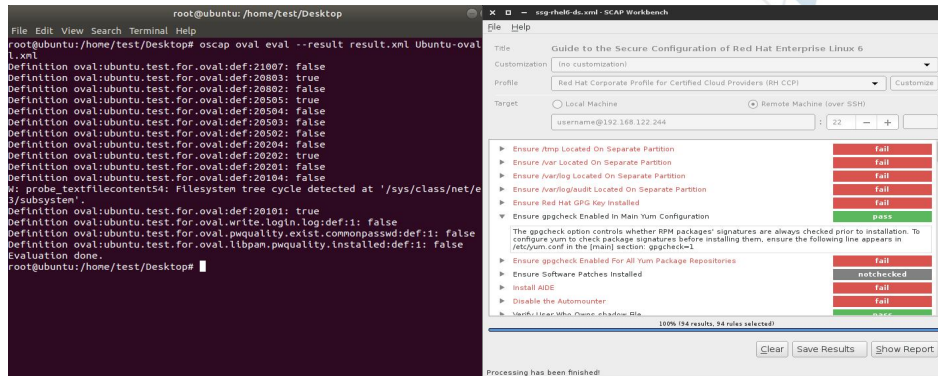
4. SCAP을 이용하는 보안 모니터링 도구 및 모듈³⁾

시스템이 알려진 위협 및 취약점에 대응하는 것은 필수적이고 지속적인 프로세스이며, 대응하기 위한

3) <https://csrc.nist.gov/Projects/scap-validation-program/validated-products-and-modules>

보안 정책을 마련하고, SCAP을 이용하여 시스템 구성요소를 점검하는 모니터링 과정이 필요하다. 이에 SCAP 보안 프로토콜의 사양을 기반으로 한 보안 자동화 모니터링 도구들이 출시되었으며, 도구들은 작성된 SCAP 문서를 기반으로 취약점 및 시스템 객체를 식별하고, 시스템의 보안 정책 준수 여부를 평가 및 보고한다.

가. OpenSCAP



[그림 3] OpenSCAP을 이용하여 테스트하는 모습

OpenSCAP은 Redhat사에서 주도하여 개발되고 있으며, SCAP 프로토콜 사양을 이용하여 시스템을 감사하기 위한 보안 스캐너로서 개발되고 있는 오픈소스 프로젝트이다⁴⁾. OpenSCAP에서 지원하는 OSCAP 프로그램은 대표적인 보안 스캐너로 시스템의 보안 구성 설정을 확인하고, SCAP 표준 및 사양을 기반으로 하는 규칙을 사용하여 시스템의 손상 징후를 검사할 수 있다. 기본적으로 XCCDF, OVAL과 같은 SCAP 컴포넌트들을 결합한 벤치마크 문서를 바탕으로 시스템의 보안 정책 준수 여부를 평가하며, OVAL 인터프리터(Interpreter)의 역할 또한 수행한다. OpenSCAP 도구는 크게 네 가지의 도구로 나뉘어 지원한다.

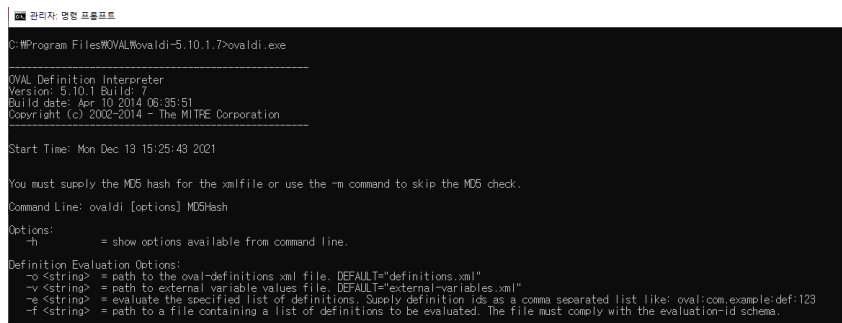
- **OpenSCAP Base** : OpenSCAP Base는 SCAP을 이용하여 시스템의 구성 및 취약성 스캔을 수행하는 커멘드라인 도구(OSCAP)이다. SCAP 표준 XML 스키마에 따라 SCAP 콘텐츠를 검증하며, 이를 바탕으로 시스템 평가를 수행한다.
- **OpenSCAP Deamon** : 백그라운드에서 실행되는 데몬 서비스이며, 지정된 일정에 따라 기계와 컨테이너를 평가한다. 특정 정책에 대해 시스템을 지속적으로 평가하거나 혹은 일회성 평가를 할 수 있다.
- **OpenSCAP WorkBench** : OpenSCAP Base보다 기능이 제한적이거나, 그래픽으로 제공되는 유틸리티이므로 쉽게 시스템을 평가할 수 있다.
- **OSCAP-SSH** : 네트워크를 통해 원격 컴퓨터를 스캔하고, 결과를 수집할 수 있다.

OpenSCAP 제품군은 기본적으로 로컬 시스템상에서만 시스템을 스캔하고 평가하므로 해당 플랫폼

4) OpenSCAP User Manual, <https://github.com/OpenSCAP/openscap/blob/maint-1.3/docs/manual/manual.adoc>

에 대한 OpenSCAP 빌드가 필요하며, OSCAP-SSH도 원격 시스템에 설치되어 있어야 하므로 OpenSCAP 내에서 평가 가능한 플랫폼이 유닉스, 리눅스 및 윈도우로 한정되는 한계점을 가진다. NIST에서 인증받은 오픈소스 프로젝트이므로 기존 OVAL 사양에 없는 새로운 테스트 추가 및 다양한 플랫폼 확장 작업이 이뤄지고 있어 현존하는 SCAP 보안 모니터링 도구 중에서 가장 발전 가능성이 큰 도구이다⁵⁾. 또한, OpenSCAP은 C언어로 작성되어 스캔 속도가 타 제품군과 비교하여 빠른 장점을 지니고 있다. 평가 결과의 경우, XML 파일 또는 HTML 보고서로 출력하여 시스템의 전반적인 평가 결과를 확인할 수 있다.

나. OVAL Interpreter (ovaldi)



```

관리자 명령 프롬프트
C:\Program Files\OVAL\ovaldi-5.10.1.7>ovaldi.exe

OVAL Definition Interpreter
Version: 5.10.1 Build: 7
Build date: Apr 10 2014 06:35:51
Copyright (c) 2002-2014 - The MITRE Corporation

Start Time: Mon Dec 13 15:25:43 2021

You must supply the MD5 hash for the xmlfile or use the -m command to skip the MD5 check.
Command Line: ovaldi [options] MD5Hash

Options:
-h          = show options available from command line.

Definition Evaluation Options:
-o <string> = path to the oval-definitions xml file. DEFAULT="definitions.xml"
-v <string> = path to external variable values file. DEFAULT="external-variables.xml"
-e <string> = evaluate the specified list of definitions. Supply definition ids as a comma separated list like: oval:com.example:def:123
-f <string> = path to a file containing a list of definitions to be evaluated. The file must comply with the evaluation-id schema.
  
```

[그림 4] OVAL 인터프리터의 옵션을 확인하는 모습

OVAL 인터프리터(ovaldi)는 MITRE사에서 개발한 프로그램으로 OVAL 커뮤니티에 OVAL 언어의 오픈소스 레퍼런스 구현을 제공하기 위해 개발된 프로젝트이며, OpenSCAP과 마찬가지로 로컬 시스템에서 보안 관련 구성 정보를 수집하고, 취약점 및 구성 문제에 대한 정보를 분석하여 분석 결과를 보고한다. 문제는 SCAP 컴포넌트 중에서 오직 OVAL만 지원하는데, OVAL 중에서도 개발 우선순위 지정으로 인해 필요에 따라 테스트에 대한 개발을 추진하여 모든 테스트를 지원하지 않는다⁶⁾. 플랫폼 또한 윈도우 및 일부 유닉스에 크게 초점이 맞춰져 있어 그 외의 플랫폼에서의 평가가 불가하며, 오픈소스 프로젝트임에도 불구하고 2014년 이후 활발한 기능 추가 작업이 이뤄지고 있지 않다.

다. jOVAL : Continuous Monitoring

jOVAL 제품군은 Java 언어로 작성되었으며, SCAP 벤치마크 문서를 이용하여 플랫폼의 기기들을 원격으로 평가할 수 있는 다중 플랫폼 구성 컴플라이언스 스캐너이다. SCAP 컴포넌트들 중에서도 OVAL의 경우 최신 버전만 지원하는 대신 해당 OVAL 언어 버전에서 지원할 수 있는 모든 테스트를 이용할 수 있는 특징이 있으며, 타 도구들과 비교하여 원격 시스템에 jOVAL이 설치되어 있지 않아도 SSH

5) Add support for FreeBSD, derived from <https://github.com/OpenSCAP/openscap/pull/1574>

6) OVAL Community, "Supported tests". <https://sourceforge.net/p/ovaldi/wiki/Supported%20tests/>

통신을 이용하여 모든 시스템 플랫폼에 대해 SCAP 평가를 원격으로 진행할 수 있는 장점이 있다.

Ubuntu-oval-all.xml scanned Sep 9, 2021 @ 10:17:01 am

9/9/21 10:17 AM

Select All Rules
Default Scoring Model with a PASS threshold of 3.
1 Target in scan: 192.168.220.132

Results Overview Rule Diagnostics Learn more about these reports.

Passed (1): 192.168.220.132

Results by Rule (13 Rules)	Reference	Impact	192.168.220.132 (23.1)
Compliance: 기관에서 도입 운영 중인 줄...	acco...	7.7	FAIL
Compliance: 인출서버 자원관리서버 가상...		7.7	FAIL
Compliance: PC 노트북 휴대용 스마트패...		7.7	Unknown
Compliance: 서버 및 패치관리시스템 관...		7.7	FAIL
Compliance: SPF(Sender Policy Framewo...		7.7	Unknown
Compliance: 서버 접속시 SSH RDP를 기...		7.7	FAIL
Compliance: 보안지원이 중단된 운영체제...	pack...	7.7	PASS
Compliance: PC 노트북 휴대용 스마트패...		7.7	FAIL
Compliance: PC 노트북 휴대용 스마트패...		7.7	Unknown
Compliance: 정보보호시스템 관리자 접속...		7.7	PASS
Compliance: 내부망PC에서만 업무자료를 ...		7.7	PASS
Compliance: 방문인 위반사항통제없는 ...		7.7	FAIL
Compliance: 정보보호시스템 대상 관리자...		7.7	Unknown

Hosts.allow FALSE oval:kr.re.nsr.oval.check.hosts.allow:tst:1
At least one item must be found.
No Items Found: Textfilecontent54 Object oval:kr.re.nsr.oval.check.hosts.allow:obj:1

Parameter	Value	Notes
filepath	/etc/hosts.allow	operation: equals datatype: string
pattern	^sshd\$?s\$([0-9]{1,3}\.){3}[0-9]{1,3}([v0-9]{1,2})?	operation: pattern match datatype: string
instance	1	operation: equals datatype: int

Hosts.deny FALSE oval:kr.re.nsr.oval.check.hosts.deny:tst:1
At least one item must be found.
No Items Found: Textfilecontent54 Object oval:kr.re.nsr.oval.check.hosts.deny:obj:1

Parameter	Value	Notes
filepath	/etc/hosts.deny	operation: equals datatype: string
pattern	^sshd\$?s\$ALL	operation: pattern match datatype: string
instance	1	operation: equals datatype: int

[그림 5] jOVAL 프로그램을 이용하여 테스트한 결과

jOVAL Professional 제품의 경우 원격을 스캔하면 SCAP 평가결과가 '\$scan_data' 디렉터리 내에 ARF(Asset Report Format) 리포트 형식으로 저장되며, ARF로 보고된 결과를 프로그램 내에 나타낸다. jOVAL 프로그램으로 HTML 리포트를 생성하지는 못하나, 대신 수집된 시스템 객체 및 평가결과를 프로그램 내의 평가결과 및 상세지표로 자세히 나타내어 평가의 근거를 쉽게 확인할 수 있다.

jOVAL 회사는 2011년 창업 이래 오픈소스를 상업화시킨 이후 SCAP의 문법 개정 및 jOVAL의 플랫폼 확장을 위해 노력해왔으나, 2021년 12월 Aritic Wolf 사에 편입되어 현재는 jOVAL의 상업적 구매 가능 여부를 알 수 없다.

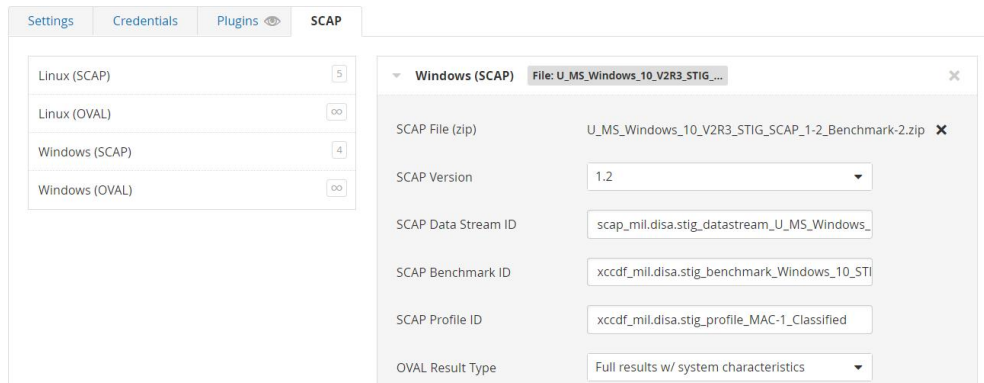
라. Nessus Professional

Nessus Professional은 SSH 통신을 통해 IT 네트워크 내 시스템들의 구성요소를 원격 스캔하고, 평가하는 웹 기반 보안 스캐너이다. SCAP 스캔을 비롯하여 멀웨어 스캔, 네트워크 스캔 등 미리 구성된 450개 이상의 시스템 스캔 템플릿(Template)을 갖추고 있어 취약점 발생 지점을 다양한 스캔을 통해 파악할 수 있고, 취약점 정보를 플러그인(Plug-in) 형태로 저장되어, 플러그인을 가져다 사용함으로써 일부 취약점 평가를 신속하게 수행할 수 있는 특징을 지니고 있다⁷⁾.

SCAP 평가의 경우 'Advanced Scan' 또는 'SCAP and OVAL Auditing' 템플릿을 이용하여 할 수 있으며, SCAP 벤치마크 문서와 OVAL XML 문서 중 하나를 택하여 평가를 진행할 수 있다. 라이브 결과를 보여줄 수 있는 장점을 지니나 SCAP 버전의 경우 SCAP 1.0~1.2 버전까지만 지원하며, Ubuntu

7) Tenable, Nessus Professional Datasheet. <https://www.content.shi.com/SHIcom/ContentAttachmentImages/SharedResources/PDFs/Tenable/Tenable-Nessus-Professional-052421-Datasheet.pdf>

커널과 같은 일부 리눅스는 해당 템플릿에서 평가를 진행할 수 없는 한계점이 존재한다. 또한, SCAP 또는 OVAL 파일을 이용하여 평가할 시 평가하고자 하는 파일을 zip 확장자로 압축하여 업로드해야 하며, SCAP 파일을 첨부했을 시 SCAP Data Stream ID, SCAP Benchmark ID, SCAP Profile ID를 일일이 지정해야 하는 불편함을 지닌다.



[그림 6] Nessus Professional를 이용하여 스캔을 진행하는 모습

정리한 SCAP 보안 모니터링 도구들의 특징을 총합하여 정리한, 대표적인 모니터링 도구들의 특징 점 비교 현황을 [표 2]에 나타냈다.

[표 2] SCAP 보안 모니터링 도구들의 기능 비교표

	OpenSCAP	ovaldi	jOVAL	Nessus
SCAP 벤치마크 문서 평가 가능 여부	O	X	O	O
원격 평가 가능 여부	△ (로컬 시스템에 설치되어야 함)	X	O	O
스캔 속도	1~5초 이내	1~5초 이내	10~20초	30초 이상
다운로드 가능 여부	O	O	X	O
리포트 가능 포맷	XML, HTML	XML, HTML	ARF	.

그 외에도 SCC (SCAP Compliance Checker) , IBM BigFix Compliance, McAfee Policy Auditor 등이 있다.

5. 결어

운영체제 및 각 소프트웨어들은 유닉스나 리눅스 계열에서는 설정파일을 가지고 있고 윈도우즈에서는 레지스터리에 설정 정보들을 포함하게 된다. 이러한 설정 정보를 이용하여 운영체제나 소프트웨어들이 갖는 취약점(일반적으로 패치인 경우)을 파악할 수 있다. 이러한 취약점들은 NCP(National Checklist Program)⁸⁾에 보고 되고 있으며, 각 목표 시스템에 따라 분류하여 XCCDF 또는 SCAP 1.3 content 형식으로 제공하고 있다. 국내에서도 SCAP과 같은 상시모니터링을 위한 체계를 갖도록 하는 연구가 필요하고 이를 통해 소프트웨어를 생산하는 기업은 취약점을 제공하고, 일반 기업에서는 정보시스템의 취약점을 모니터링할 수 있는 체계를 통해 상시 보안 취약점에 대한 적극적인 대책을 강구할 수 있도록 하는 제도적인 변화가 요구된다.

8) <https://ncp.nist.gov/repository>

발행일	2022년 1월
발행처	한국인터넷진흥원 (전라남도 나주시 진흥길 9)
기획	한국인터넷진흥원 미래정책연구실 정책분석팀
편집	(주) 해리