The Open Web Application Security Project

VulnXML
Proof of Concept
Vision Document

Version 1.1

Revision History

Date	Version	Description	Author
7/5/2002	1.0	First Draft	Mark Curphey
July 8, 2002	1.1	Comments from Jennifer	Jennifer Tharp

Table of Contents

Vision Document

1. Introduction

This document sets out the high level vision for the VulnXML "proof of concept" project. VulnXML aims to make free web application security knowledge available to everyone and anyone at the same time.

2. Positioning

2.1 Problem Statement

The problem of	Like many other parts of the IT industry, the information security industry has developed extremely fast with few standards bodies and very little co-operation and co-ordination between vendors and the user community.
the impact of which is	This has lead to two specific problems. When security researchers publish security advisories or vulnerabilities, they either do so in an ambiguous textual form or using a proprietary data format for use in their tools. This net effect is that security data has become tightly coupled to specific tools and cannot easily be shared across different tools.
a successful solution would be	If an independent body created an open standard data format from which anyone could describe web application security vulnerabilities, security knowledge could be openly shared with both tools and users. Such a common open format would enable researchers to publish vulnerabilities that could be then used to produce checks in a variety of tools that understand the format. This would allow a researcher to publish a vulnerability once and enable it to quickly be able to be consumed by tools (and users) irrelevant of the specific technology the end user choose to use. Note: This format is for static known vulnerabilities only and would not be used to find dynamic unknown security issues. That is to say it will describe a specific discrete issue and not a class of issues.

2.2 Position Statement

The VulnXML will create an open standard format for web application security vulnerabilities only. Whilst we believe it could be extended to other classes of security problems, they are beyond the scope of this project.

3. Stakeholder and User Descriptions

3.1 Stakeholder Summary

Name	Nominations	Responsibilities
VulnXML Designers	OWASP -Rogan Dawes -Mark Curphey Kavado -Yuval Ben-Itzak	The VulnXML designer is responsible for designing and maintaining the data format, liaising and working with the other stakeholders to ensure that it meets their requirements. They will also be responsible for creating the initial conversation candidate list.
Security Researchers	TBD	The security researchers will be responsible for converting the candidate lists into the VulnXML format, as well as adding new vulnerabilities in the format as they are published.
Web Application Security Scanner Developers	Kavado ScanDo -Yuval Ben-Itzak OWASP WebScarab -Tim Panton -Ingo Struck -Steve Taylor	The web application scanner developers will be responsible for ensuring the format contains the necessary meta data to allow the developer to construct a parser that will be capable of building checks.

3.2 User Summary

Name	Description	Responsibilities	Stakeholder
Web Application Scanner Users	These users are typically security professionals or consultants who undertake web application security assessments.	Defines end user needs	The security researchers are will act as stakeholders for these users.

3.3 User Environment

Only tools that can understand the VulnXML format will be able to use the checks written in this format. Kavado's ScanDo product (a commercial web application scanner) and OWASP's WebScarab (an open source free web application scanner) will have parsers developed as part of this proof of concept project. We may additionally provide a standalone tool written in Perl that can use VulnXML checks.

3.4 Alternatives and Political Landscape

As far as we are aware this has not been done before, and as such, we are viewing this project as a proof of concept project.

We are aware that this project will not be well received among some web application scanner developers. As well as proprietary research, these companies often take publicly released advisories and create proprietary checks that they then sell to their clients as part of the software maintenance for their tools (provision will be made for copyright information). It is entirely possible that an alternative format will be proposed by some of those vendors or individuals that more tightly meet their commercial goals and would enable a limited sharing model. We, however, think that when WebScarab is released and the general user community has access to use these open checks in the tool of their choice, be it commercial or open source, then the database of checks and the number of security researchers writing checks will create a significant enough momentum for others to adopt the checks. After all, the inability to accept a large number of accurate, up-to-date checks would put a vendor at a competitive disadvantage.

CVE and the Bugtraq databases – The common Vulnerabilities and Exposures (CVE) database and the Bugtraq database do an excellent job of capturing, recording and classifying security vulnerabilities. They are not, however, designed to capture sufficient information about a web application security vulnerability that would enable it to be automatically built into a check that a tool could use. We will be making every effort to reference CVE meta-data of any vulnerability we convert to the VulnXML format and have made provision in the initial data type definition.

4. Product Overview

The VulnXML format will be an open source and openly published standard XML document data type definition from which users can describe a particular security vulnerability in a web application in an unambiguous manner. The data type definition (DTD) will allow the security check developer or security researcher to describe enough meta-data about the vulnerability that an automated program could build an http request or series of requests to determine if the vulnerability exists on the system being tested.

NB: There will be no requirements on a security researcher to release his checks described in this format to the public. This protects client confidentiality agreements and intellectual property.

Whilst the data type definition is the primary deliverable for this project, in order to demonstrate how this format will work we acknowledge that we will also have to build, maintain and make available content in this format as well as reference implementations and working tools that will user the format. Therefore we will develop a vulnerability database that will be made available to the public for no charge and work with the Kavado ScanDo and OWASP WebScarab teams to demonstrate how two tools (commercial and open source) can share the same publicly available data.

In summary the products are:

- Data Type Definition
- Publicly Available Vulnerability Database
- Proof of Concept Tools

4.1 Product Perspective

The data type definition would be used in conjunction with a tool or component that parses the data and build the checks for the specific tool in question. Whilst we will provide a reference implementation, each vendor that wishes to use the format will also need to build a parser. Each parser will be unique, as specific products will build tests in a unique way.

4.2 Assumptions and Dependencies

We are assuming that;

- Tools vendors will want to incorporate as many checks as possible in their tools
- Security researchers want their advisories to be translated into security checks in tools
- Users want to have access to the latest security research
- The community will get behind the VulnXML format

5. Product Features

VulnXML must be:

- Standards based
- Free to use
- Open to everyone
- Accessible to everyone
- Independently controlled and not vendor (commercially) biased
- Platform agnostic
- Able to allow contributors to maintain copyright and intellectual property
- Able to describe all static web application security vulnerabilities
- Extensible