

An S-vector for Web Application Security Management

Russell R. Barton
Smeal College of Business
Administration
Penn State University
University Park, PA 16802
001-814-865-7669
rbarton@psu.edu

William J. Hery
Penn State eBusiness Research
Center
51 Mt. Pleasant Road
Morristown, NJ 07960
001-973-895-3451
hery@nac.net

Peng Liu
School of Information Sciences and
Technology
Penn State University
University Park, PA 16802
001-814-863-0641
pliu@ist.psu.edu

ABSTRACT

Existing security scoring methods are expensive to implement, lack management orientation and are “best practice” based, and thus have only transient meaning. This paper proposes a web application security assessment method based on a security scoring vector (S-vector). The S-vector assessment method would be used by IT administrators to manage security of their web applications. It shares some analogous features with the R-value for insulation.

Categories and Subject Descriptors

H.3.5 [Online Information Services]: Commercial services and Web-based services. K.6 [Management of Computing and Information Systems]: Software management, Security and protection.

General Terms

Management, Measurement, Security, Standardization, Verification.

Keywords

Security, web application, S-vector.

1. INTRODUCTION

Web applications (as distinguished from browsers or server software such as Apache) themselves have recently become the subject of attack. The focus of these attacks are on the custom application code, which can be accessed via the web. The applications can contain passwords or hints, local path information, back end server names and IP addresses and SQL queries and passwords. Attack types relate to authentication, session management, database interaction and generic input validation (GET and POST). Attacks are through SQL injection, cross-site scripting, misuse of hidden tags, exploitation of server-

side includes and misuse of the ability to append information to files [5]. The introduction of wireless interfaces to web applications (to support handhelds and laptops) further complicates the security problem.

The Commonwealth of Pennsylvania has over 200 web applications that allow it to conduct business more effectively. The “PAPowerPort” allows residents to renew a driver’s license/vehicle registration, obtain tax forms, purchase hunting/fishing licenses, learn about state parks, file worker’s compensation employer’s wage reports, etc. Managing the risk assessments for these applications requires the efforts of internal reviewers and contractor personnel. Prior to launch, these reviewers must approve an Electronic Commerce Security Assessment (ECSA) for each application. The Commonwealth is interested in a practical method for scoring the security-worthiness of these applications. Such a system would allow them to answer questions such as:

- What is the security status of each web application?
- What are the security requirements for each application (i. e., what is “good enough” security)?
- How do you trade off between maximal security and a pragmatic, affordable approach in application development, acquisition, and deployment?
- Where are your vulnerabilities, and how do you know that they are weak?
- How will you know whether your modifications have significantly improved security?
- How do you prioritize security enhancement projects?
- What security measures do you have? How do you know they are valid?

Existing security scoring methods are expensive to implement, lack management orientation and are “best practice” based (and thus have only transient meaning). As an alternative, this paper suggests a web application security assessment method based on a security scoring vector (S-vector). The S-vector assessment method would be used by IT administrators to manage security of their web applications. It shares some analogous features with the R-value for insulation, developed at Penn State [8]. Like the R-value, it is designed to be used by non-expert decision makers. Like the R-value, the S-vector score must be compared with requirements to judge the adequacy (attic insulation requirements

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Conference '00, Month 1-2, 2000, City, State.

Copyright 2000 ACM 1-58113-000-0/00/0000...\$5.00.

are higher than garage door requirements). Unlike the R-value, this characterization will not consist of a single number, but rather a set of characterizations along a number of security dimensions. Key attributes of our approach include a management focus, ease of use and clarity of results for non-experts, and use by application owners with inputs from developers and vendors.

In many large organizations (such as the Commonwealth), large numbers of web applications are “owned” by the departments that provide the related services, but the applications are clustered on a smaller number of centrally managed servers. These centrally managed services include security management of the servers (e. g., OS patching, OS level auditing) and the internal networks (e. g., firewalls, intrusion detection systems). In this study, we focus only on the specific security needs at the application level, and on system vulnerabilities that may be introduced by the applications. These needs may vary greatly among the applications: for example, on-line driver’s license applications have a much greater need for confidentiality of citizen information and authentication of users than state park information web applications. The methodologies we propose to develop are to be used by the application “owners” and must function within the context of their business practices.

This report describes a research project that is in its early stages.

The next section defines possible elements of the S-vector and describes a web application security management system in which it might be used. The following section describes how this approach compares with existing security assessment methods, and how it addresses known difficulties in the development and use of security scoring systems. The final section identifies the research questions to be answered.

2. S-VECTOR DESCRIPTION

2.1 Scope

The S-vector is intended for limited scope: security management for web applications. That is, the security management of custom software written using HTML, XML, Java etc. to allow web browser interaction to effect a business or government function. While the focus is on the security assessment of the web applications themselves, an effective assessment will require some attention to the external environment, including connected database servers and internal users. The security assessment must consider:

- the web application software,
- the software development process that was used to create the web application,
- the configuration management and operating policies affecting the web application and server operating environment, and
- the business processes that the application supports.

2.2 Possible Elements

We believe that the components of the S-vector will fall into three major categories.

1. The *technical capabilities* components of the S-vector. These capabilities will include authentication technology, encryption methodology, access control methods, and others. These capabilities essentially describe the strength and (in some cases) methodology to achieve application security, and generally can be objectively determined.
2. The *structural protection* components of the S-vector. The structure of some web applications can provide additional protection against malicious actions even if there are security compromises at lower levels in the system. For example, integrity checking in the application code can be used to prevent attacks that rely on security bugs in the operating system. These components in the S-vector indicate the use or absence of each of a list of such structures.
3. The *procedural methods* components of the S-vector. These methods include software development methodology (including testing and static verification tools and techniques), system management, security policies, etc. These methods cover the security related procedures used in development, installation, use, and management of the application that give a level of assurance that the technical capabilities have been achieved and are maintained in deployment. They may be subjectively determined.

2.3 Mapping Security Requirements to a Target S-vector

Table 1 shows a typical strategy to use security-related information to define the security level that must be achieved for a particular web application. Security classes are identified by the coding in the table: class A web applications have lowest security requirements, class C applications have the highest. For example, if the threat of security compromise is ‘high,’ and the impact of such a compromise would be ‘medium,’ then the application should meet level ‘B’ security requirements. This approach has important shortcomings. First, there are many dimensions to security, and it is hard to capture the requirements through a single label, ‘B.’ Second, if the a web application falls short of the required ‘B’ level, what actions are recommended? Third, for this particular example, very few applications fall in the high and low categories, the classification cannot be used to prioritize attention for security improvements among the bulk of the web applications receiving a ‘B’ requirement.

The practicality of the S-vector requires the ability to take security-related information connected to the business process supported by the web application (including risk, threat, policy and legal requirements) and use it to construct *target* S-vector elements. In this way, the security score S-vector for a web application can be compared to the target S-vector, and elements that fall short can be addressed. The advantage of a vector security score and target is that a gap in one coordinate of the S-vector identifies the nature of the security shortfall, permitting management to focus on actions that will address the security need.

Table 1. Typical security requirements classification.

		Threat		
		L	M	H
Impact	H	B	B	C
	L	A	A	A

	M	B	B	B
	L	A	B	B

The S-vector elements should be developed in conjunction with the development of the security requirements mapping, since there is no point in developing an element of the S-vector if a target level cannot be established.

2.4 Operation of the S-vector Strategy in a Web Application Management System

Figure 1 shows the general structure for a web application management system in which the S-vector method would play a role. A periodic assessment of web applications yields an S-vector for each application. New applications receive an initial S-vector through the same process. Security requirements are determined by impact and threat, in conjunction with government policy and legal requirements. These must be mapped into a target or requirement vector that parallels the S-vector. Management uses the application S-vectors in conjunction with security requirements to determine web applications with high priority for the attention of presumably scarce resources for addressing security concerns.

2.5 Example

[Example to be provided at the workshop and in the final version of this report.]

3. EXISTING ASSESSMENT METHODS

3.1 Brief Descriptions

Currently, there are programs and initiatives, both in the private and public sectors that have sought to provide tools for managing computer system security. The focus has been on either software or hardware modules, or overall system security.

The first widely used rating system was the Department of Defense Trusted Computer Evaluation Criteria, DoD Standard DOD 5200.28-STD [11]. The goal of this “Orange Book” standard was to protect the confidentiality of classified information on a system with multiple users having potentially different access rights to various pieces of information. The Orange Book defined seven levels of “trusted computers,” ranging from D (Minimal Security) to A1 (Verified Design). For these standards, both capability and assurance are combined in the level of trust: a higher rating not only has stronger security capabilities, but has greater assurance that those capabilities are met. Evaluation is done by the DoD, and is a very lengthy and expensive process even at the lowest level.

The Orange Book focuses on individual, multi-user computers. Over time, the principles were extended to other settings. One such widely used standard is the “Red Book,” the Trusted Network Interpretation (TNI) of the Trusted Computer System Evaluation Criteria (TSEC). This document extends the rating system to networking elements, such as routers.

In December of 2000, the International Standards Organization published ISO 17799, a comprehensive set of controls that comprises best practices in information security. This internationally recognized security standard is intended to identify a range of controls necessary for most situations where

information systems are used in industry and commerce infrastructures [2].

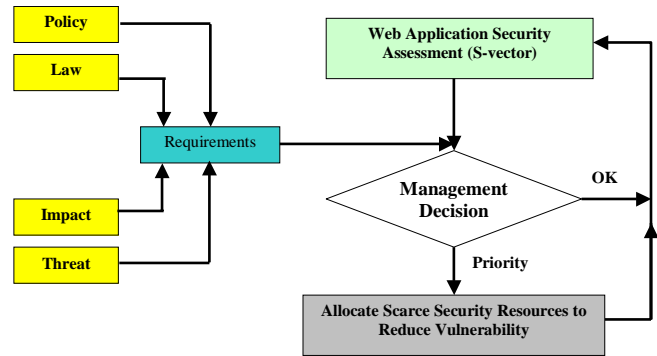


Figure 1. Operation of a web application security assessment system using an S-vector.

The Common Criteria ISO/IEC 15408 specifies security requirements for the development of products and systems with IT security functions [1]. It is an international effort of organizations from Canada, France, Germany, the Netherlands, the United Kingdom, and the United States. This approach requires extensive (and consequently expensive) assessment activities that are not designed for use with individual web applications.

The Operationally Critical Threat, Asset, and Vulnerability Evaluation (OCTAVE) Information Security Risk Evaluation was developed by Carnegie Mellon’s Software Engineering Institute. It is an approach for self-directed security risk evaluations that “puts organizations in charge, balances critical information assets, business needs, threats, and vulnerabilities, measures the organization against known or accepted good security practice, and establishes an organization-wide protection strategy and information security risk mitigation plans” [10].

The Systems Security Engineering Capability Maturity Model (SSE-CMM) assesses the maturity of an organization’s information security processes [3]. The focus is on the software design and maintenance process, rather than on the software/hardware itself. The SSE-CMM allows organizations to evaluate security engineering practices and identify areas for improvement. It might be used in conjunction with conformance to standards specified by any of the above methods. This characterization is an important part of web application security assessment, and is incorporated in the S-vector strategy.

The Federal Information Technology Security Assessment Framework gives five levels of IT security program effectiveness: 1: documented policy, 2: documented procedures, 3: implemented procedures and controls, 4: tested and reviewed procedures and controls, 5: fully integrated procedures and controls [6], [7]. Like the SSE-CMM model, its focus is on the effectiveness of the security program, rather than the applications themselves.

3.2 Comparison of the S-vector Method

The S-vector strategy differs from past developments in security assessment. Table 2 shows a comparison of the S-vector approach with some of these methods. The S-vector approach has

a narrower focus than the others: web applications only, rather than system assessment or an assessment of a collection of components. This narrow focus gives hope that scoring and target-setting procedures can be developed.

Nonetheless, the S-vector strategy should take advantage of developments in existing assessment methods. One case in particular is SSE-CMM. SSE-CMM identifies a number of areas for assessment, including access control, accountability, audit, authentication, availability, configuration management, detection, education and awareness, monitoring, policy and procedures, standard design processes; use metrics, frequency of regular audit reviews, percent of users with passwords meeting policy, number of failed login attempts, and frequency and compliance with virus detection updates [4].

Along with other differences from existing security scoring methodologies, the primary uniqueness of S-Vector is that it is *management-oriented*, instead of being security capability oriented (e.g., CC), or security process oriented (e.g., SSE-

CMM). In particular, (a) S-Vector is driven by the management needs of executives. What are the management needs of executives? The executives know that “perfect” security is either not achievable or not cost-effective, hence, they are looking for “good enough security”. For this purpose, they need to exactly know “how much security is good enough?” Existing methods used to answer this question are ad hoc, qualitative, and based on best practice. The goal of S-Vector is to help executives to answer this question systematically and quantitatively, and to deploy and maintain “good enough security” for web applications.

3.3 Criticisms and Responses

The S-vector strategy addresses a number of criticisms leveled against security scoring schemes. Schneier [9] cites seven problems with a UL-type security rating. Each is addressed by the S-vector approach that we have described.

Table 2. A Comparison of Security Assessment Methods.

Category	Security Assessment Method				
	<i>S-vector Approach</i>	<i>Common Criteria</i>	<i>COBRA</i>	<i>OCTAVE</i>	<i>SSE-CMM</i>
Status	Proposed research	Established, evolving standard	Available Commercial Product	Documented Methodology with training available	Documented Methodology with training available
Level of analysis	Application level (e. g., vehicle registration renewal)	Typically, system component level (e. g., firewall, operating system, DBMS) or security specific application	Overall system	Overall system	Overall system
Focus	Management needs: score vs. requirement	Meeting CC protection profile	Best practices	Best practices	Best practices
Requirements, Threat, and Impact Assessment	By application. Based on questionnaire.	Requirements in Protection Profile. Impact Assessment not included.	By overall system. Automated tool based on surveys and expert systems	By major asset class.	Not included.
Technical capabilities (confidentiality, integrity, authentication, availability, etc.)	Numerical scores for components, with an appropriate subset used for each application.	Fixed protection profiles defined by Common Criteria or vendor for each class of system component.	Includes a subset of technical capabilities implemented by the system infrastructure as a whole.	Capabilities for overall infrastructure (hardware, OS, network).	Indirectly only: SSE-CMM addresses the processes used to implement and support them.
Structural protection	S-vector includes explicit reference to relevant structures	Such structures may implicitly influence the EAL evaluation for components.	Not included.	Not included.	Indirectly only.
Procedural methods: development and testing	Included as S-vector components.	Included in EAL evaluation for components.	Yes	Not included.	Yes
Procedural methods: management and field use	Included as S-vector components.	Not included.	Yes	Not included.	Yes
Independent Testing	During research phase only.	Uses more rigorous third party evaluation methodology.	Not included.	Not included.	Indirectly only

1. "Security assessment is a moving target. There are new vulnerabilities, new attacks, new countermeasures. Any rating is likely to become obsolete within months." An S-vector approach might address this concern: i) the management-based assessment using risk of compromise is not tied directly to the

observance of a single attack event, and ii) periodic reevaluation is practical with an inexpensive assessment method.

2. "It is much too hard to test network security, due to interactions among various components of the system. Assessment would

take millions of dollars because modern software is very complex.” The S-vector strategy has limited focus: allocating security enhancement resources across many web applications. The scope makes the identification and assessment of interactions more manageable, since only web applications code will be examined, and interactions between web application features and the system need to be examined.

3. “A 30-minute rating for safecracking is meaningful. What does a 9 mean for a security rating scale of 1-10?” An S-vector based on management needs means that the elements and values will be designed to have meaning for management decision making.
4. “Failures are not always obvious.” For the S-vector, management-based assessment using risk of compromise is not tied directly to the observance of a single attack event.
5. “Security depends on context and environment.” Context can be captured by workflow analysis and by the classification of a small number of general classes of web applications. The S-vector need only characterize interactions at the web application level, which should be easier than examining all lower-level interactions, particularly since in many instances all of the web applications in the management set will operate in the same environment.
6. “[Security assessment] needs to be combined with security practices: how the user configures, people’s behavior.” The S-vector assessment includes process characterization.
7. “Assessment will be so expensive, only major suppliers could afford to have their products tested.” This remains a concern. The S-vector strategy must be developed as a practical management tool. The limited (web application) scope reduces the effort required to characterize the security.

Bennet Yee [12] has criticisms of security metrology approaches similar to Schneier. He argues the need for a multi-component value-based assessment whose values depend on the application setting of the software:

“Security has many dimensions, as described in the previous section, but it is not a simple ‘yes you have it’ or ‘no you don’t,’ even at the level of individual dimensions. There is a broad range of possible security strengths, usually with higher costs.” This supports a vector-based scoring strategy.

4. RESEARCH NEEDS

The S-vector strategy described here is simply that - a strategy, not a well-defined methodology. Research is needed to:

- Determining the elements of an S-vector.
- Defining the process to map the security requirements for a web application into a target S-vector.
- Defining the process to determine the score (S-vector) for a web application.
- Developing a management-friendly way to present the results of scoring and targeting for a set of web applications.
- Assessing the performance of the S-vector methodology.

5. ACKNOWLEDGMENTS/COAUTHORS

This paper was the joint work of many co-authors, all of whom could not be listed on the title page: Stan Aungst, Todd Bacastow, Brian Cudnik, Lee Giles, Akhil Kumar and Nirmal Pal, the Pennsylvania State University, Nasir Memon and Gleb Naumovich, The Polytechnic University, Sandra Mateer, the Commonwealth of Pennsylvania, and Dan Pantaleo, SAP America. This work was supported in part by NSF Grant DMI-0335720.

6. REFERENCES

- [1] CCPSO. Common Criteria. Produced by the Common Criteria Project Sponsoring Organizations. <http://www.commoncriteria.org/cc/cc.html>.
- [2] ISO. ISO 17799 Made Easy. ISO, 2003. <http://www.iso-17799-security-world.co.uk/what.htm>.
- [3] ISSEA. SSE-CMM Appraisal Method V2.0. ISSEA, 2002. <http://www.sse-cmm.org/lib/lib.htm>.
- [4] Jelen, G. Systems security engineering capability maturity model (SSE-CMM) profiles, assurance and metrics (PAM) working group. In Approaches to Measuring Security, NIST workshop held June 13-14, 2000. <http://csrc.nist.gov/csspab/june13-15/sec-metrics.html>.
- [5] McClure, S., Scambray, J. and Kurtz, G. Hacking Exposed: Network Security Secrets & Solutions, Fourth Edition. McGraw-Hill Osborne Media, New York, NY, 2003.
- [6] NIST. Federal information technology security framework. National Institute of Standards and Technology, Gaithersburg, MD, 2000. http://www.cio.gov/documents/federal_it_security_assessment_framework.html.
- [7] NIST. Security Self Assessment Guide for Information Technology Systems. National Institute of Standards and Technology, Gaithersburg, MD, NIST Special Document 880-26, 2001.
- [8] R-Values. Online Historical Marker Guide, The Penn State Alumni Association. <http://www.alumni.psu.edu/vrpennstate/HistMrkr/Index.html>.
- [9] Schneier, B. A cyber UL? Crypto-Gram, January 15, 2001. Published online by Counterpane Internet Security, Inc., 2001. <http://www.counterpane.com>.
- [10] SEI. OCTAVE Information Security Risk Evaluation. Carnegie Mellon University Software Engineering Institute, 2003. <http://www.cert.org/octave/>.
- [11] U.S. Department of Defense. Department of Defense Trusted Computer System Evaluation Criteria, DOD 5200.28-STD, U.S. Department of Defense, Washington, DC, December 1985.
- [12] Yee, B. Security metrology and the Monty Hall problem. Position paper, Workshop on Information-Security-System Rating and Ranking. <http://www.acsac.org/measurement/proceedings/wissr1-proceedings.pdf>, preprint dated April 2, 2001 available online at <http://www.cs.ucsd.edu/users/bsy/papers.html>.

