11/08

OWASP
The Open Web Application Security Project

OWASP Application Security Verification Standard 2008

– Web Application Edition

beta

FOREWORD

This document defines four levels of application security verification for web applications. Each level includes a set of requirements for verifying the effectiveness of security controls that protect web applications.

The requirements were developed with the following objectives in mind:

- Provide web application developers with a yardstick with which to assess the degree of trust that can be placed in their web applications,
- Provide guidance to security control developers as to what to build into controls in order to satisfy web application security requirements, and
- Provide a basis for specifying web application security verification requirements in contracts.

The requirements were designed to meet the above objectives by ensuring validation of how security controls are designed, implemented, and used by a web application. The requirements ensure that the security controls used by a web application operate using a deny-by-default strategy, are centralized, are located on the server side, and are all used where necessary.

TABLE OF CONTENTS

TABLES

TABLES

INTRODUCTION

The Open Web Application Security Project (OWASP) is an open community dedicated to enabling organizations to develop, purchase, and maintain applications that can be trusted. All of the OWASP tools, documents, forums, and chapters are free and open to anyone interested in improving application security. We advocate approaching application security as a people, process, and technology problem because the most effective approaches to application security include improvements in all of these areas. We can be found at www.owasp.org

OWASP is a new kind of organization. Our freedom from commercial pressures allows us to provide unbiased, practical, cost-effective information about application security. OWASP is not affiliated with any technology company, although we support the informed use of commercial security technology. Similar to many open-source software projects, OWASP produces many types of materials in a collaborative, open way. The OWASP Foundation is a not-for-profit entity that ensures the project's long-term success.

The primary aim of the OWASP Application Security Verification Standard (ASVS) Project is to normalize the range in the coverage and level of rigor available in the market when it comes to performing application security verification using a commercially-workable open standard. This standard can be used to establish a level of confidence in the security of web applications.

APPROACH

The OWASP ASVS defines verification and documentation requirements that are grouped on the basis of related coverage and level of rigor. Web application security verification is performed from a logical point of view by following (or attempting to follow) paths into and out of the application and performing analysis along those paths. The Standard defines four hierarchical levels (e.g. Level 2 requires more coverage and rigor than Level 1) as depicted in the figure below.

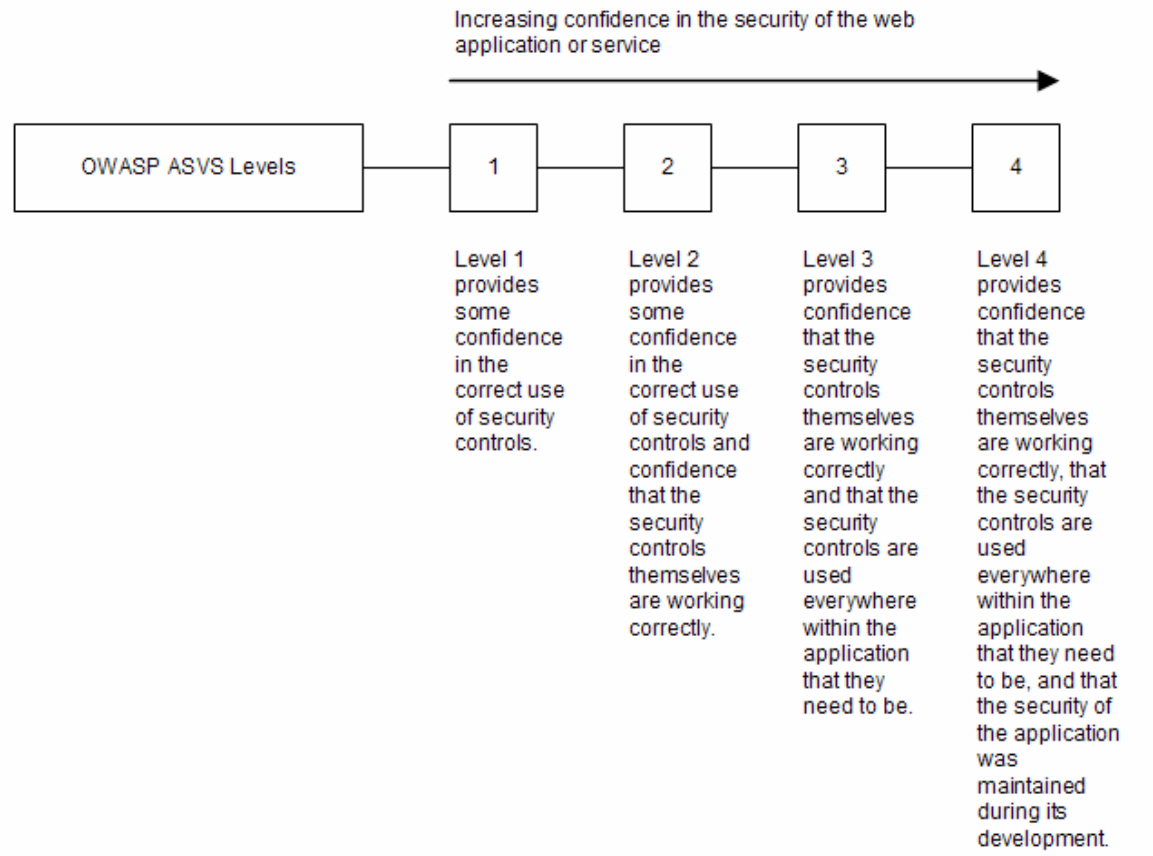

Figure 1 – OWASP ASVS Levels

The Standard further defines constituent components for Levels 1 and 2 (e.g. verification at Level 1 requires meeting both Level 1A and 1B requirements). Applications may claim compliance to either Level 1A or 1B instead of Level 1, but making such claims is weaker than claiming Level 1. Similarly, applications may claim compliance to either Level 2A or 2B instead of Level 2, but making such claims is weaker than claiming Level 2.

Verification and documentation requirements are defined in this Standard using three types of requirements:

1) Level requirements,

2) Derived Verification requirements, and

3) Derived Reporting requirements.

Level requirements define high-level web application implementation and verification requirements according to OWASP ASVS. Derived Verification requirements define low-level web application implementation and verification requirements (i.e. specific items to verify). Derived Reporting requirements define how the results of performing a web application verification according to the OWASP ASVS must be documented. The relationship between these types of requirements is depicted in the figure below.

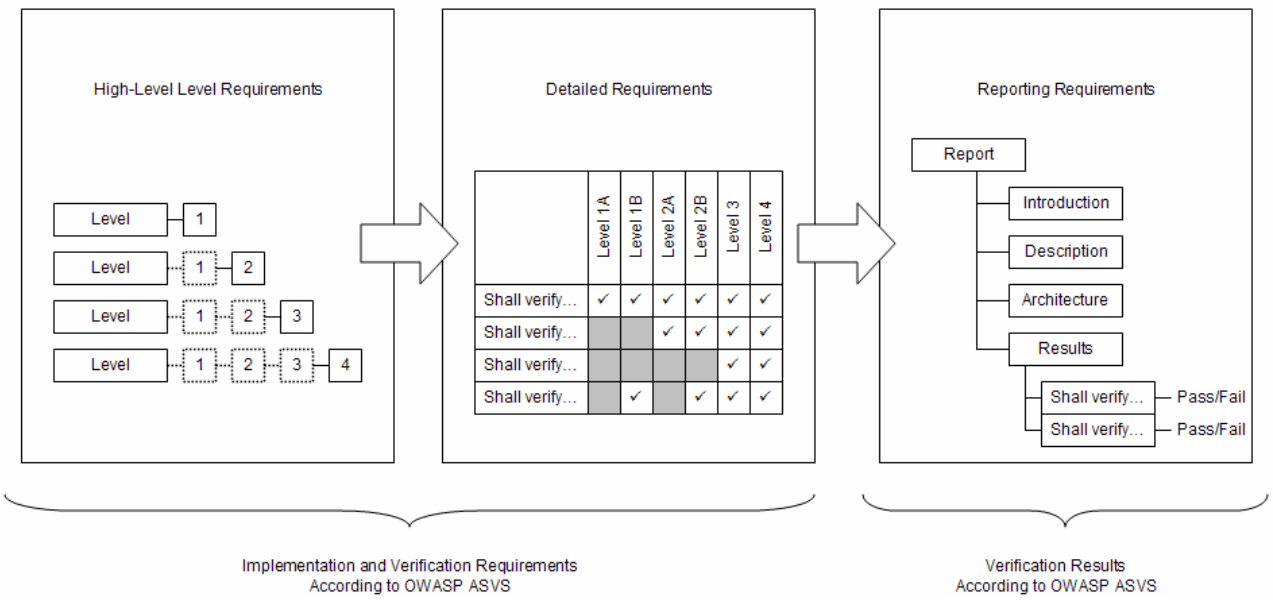

Figure 2 – Relationship between OWASP ASVS Requirements

ACKNOWLEDGEMENTS

We thank the OWASP Foundation for sponsoring the OWASP Application Security Verification Standard Project during the OWASP Summer of Code 2008. The project is led by Mike Boberski (boberski_michael@bah.com).

Project Lead: Mike Boberski (Booz Allen Hamilton)

Authors: Mike Boberski (Booz Allen Hamilton), Jeff Williams (Aspect Security, Dave Wichers (Aspect Security)

Acknowledgement is given for the contributions of: Pierre Parrend, who acted as an OWASP Summer of Code 2008 Reviewer; and finally, thanks are given to the application security verification community and others interested in trusted web computing for their enthusiastic advice and assistance throughout this effort.

APPLICATION SECURITY VERIFICATION LEVELS

The OWASP Application Security Verification Standard defines four levels of verification that increase in both breadth and depth as one moves up the levels. The breadth is defined in each level by a set of security requirements that must be addressed. The depth of the verification is defined by the approach and level of rigor required in verifying each security requirement.

It is a verifier's responsibility to determine if an application meets all of the requirements at the level targeted by a review. If the application meets all the requirements for that level, then it can be considered an OWASP ASVS Level N application, where N is the verification level that application complied with. If the application does not meet all the requirements for a particular level, but does meet all the requirements for a lower level of this standard, then it can be considered to have passed that level of verification. This standard uses the term the 'verifier' to indicate the person or team that is reviewing the application against these requirements.

LEVEL 1 – AUTOMATED VERIFICATION

Level 1 ("Automated Verification") is typically appropriate for minimum risk applications, where some confidence in the correct use of security controls is required, but the threats to security are not viewed as serious.

In Level 1, the verification involves the use of automated tools augmented with manual verification. Because automated tools generally use vulnerability signatures to find problems, this level only provides partial web application security verification coverage. The manual verification is not intended to make the web application security verification performed at this level complete, only to verify that each automated finding is correct and not a false positive.

There are two constituent components for Level 1. Level 1A is for the use of automated vulnerability scanning (dynamic analysis) tools, and Level 1B is for the use of automated source code scanning (static analysis) tools. Verification efforts may use either of these components individually, or may perform a combination of these approaches to achieve a complete Level 1 rating. The structure of these levels is depicted in the figure below.

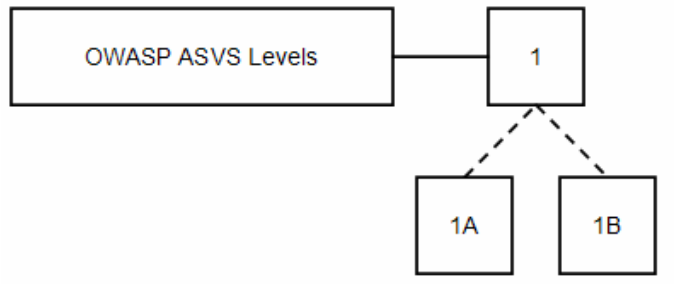

Figure 3 – OWASP ASVS Levels 1, 1A, and 1B

While it may be determined that an application meets either Level 1A or 1B, neither of these levels alone provide the same levels of rigor or coverage as an application that meets Level 1. An application that meets Level 1 must meet both Level 1A and 1B requirements.

The following are minimal requirements for Level 1, 1A, or 1B web applications:

Security Control Behavior Requirements

None: There are no requirements for how web application security controls make decisions at Level 1.

Security Control Use Requirements

None: There are no requirements for where web application security controls are used within the application at Level 1.

Security Control Implementation Requirements

None: There are no requirements for how web application security controls are built at Level 1.

Security Control Verification Requirements

L1.1: The verifier shall dynamically scan the web application according to the Level 1A requirements in the “Detailed Verification Requirements” section.

L1.2: The verifier shall perform source code scanning on the web application according to the Level 1B requirements in the “Detailed Verification Requirements” section.

Requirements that allow the use of either technique do not have to be verified with both. These verification requirements can be verified with either technique at Level 1.

Documentation Requirements

L1.3: The verifier shall create a verification report that details the web application security architecture, and the results of the verification according to the requirements in section “Verification Reporting Requirements”.

At Level 1, the web application can be defined by simply listing its components. Components may be defined in terms of either individual or groups of source files, libraries, and/or executables, as depicted in the figure below. At Level 1, the list need not be sorted or otherwise organized; the web application can be treated as groups of components within a single monolithic entity. The path or paths a given end user request may take within the application do not need to be identified and documented.

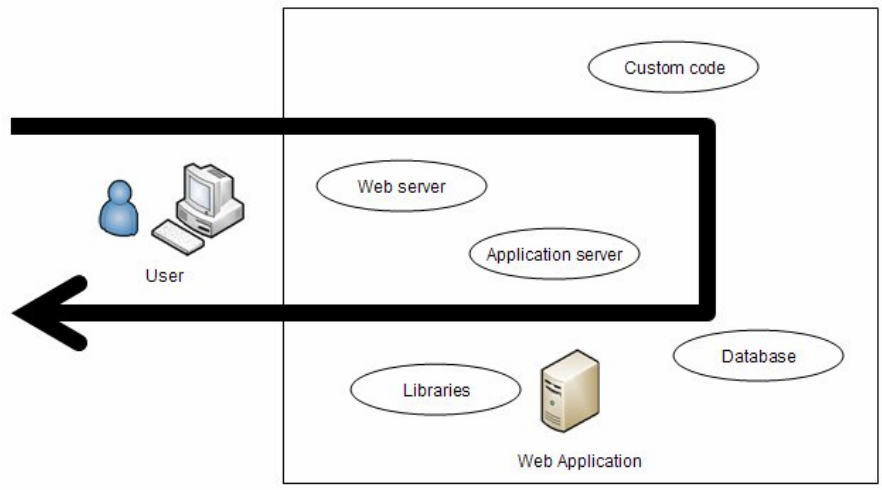

Figure 4 – OWASP ASVS Level 1 Security Architecture Example

LEVEL 1A – DYNAMIC SCAN (PARTIAL AUTOMATED VERIFICATION)

Dynamic Scanning Security Control Verification Requirements

Dynamic scanning (a.k.a. "vulnerability scanning") consists of using automated tools to access web application interfaces while the web application is running in order to detect vulnerabilities in the web application's security controls. Note that this is not sufficient to verify the correct design, implementation, and use of a security control, but is acceptable verification at Level 1.

L1A.1: The verifier shall dynamically scan the web application according to the Level 1A requirements specified in the "Detailed Verification Requirements" section.

L1A.2: The verifier shall verify all dynamic scan results using either manual penetration testing or code review. Unverified automated results are not considered to provide any assurance and are not sufficient to qualify for Level 1.

Multiple instances of a vulnerability pattern that can be traced to a single root cause should be combined into a single risk.

LEVEL 1B – SOURCE CODE SCAN (PARTIAL AUTOMATED VERIFICATION)

Source Code Scanning Security Control Verification Requirements

Source code scanning (a.k.a. "static analysis") consists of using automated tools to search through web application source code to find patterns that represent vulnerabilities. Note that this is not sufficient to verify the correct design, implementation, and use of a security control, but is acceptable verification at Level 1.

L1B.1: The verifier shall perform source code scanning on the web application according to the Level 1B requirements specified in the "Detailed Verification Requirements" section.

L1B.2: The verifier shall verify all source code scan results using either manual penetration testing or code review. Unverified automated results are not considered to provide any assurance and are not sufficient to qualify for Level 1.

Multiple instances of a vulnerability pattern that can be traced to a single root cause should be combined into a single risk.

LEVEL 2 – MANUAL VERIFICATION

Level 2 ("Manual Verification") is typically appropriate for applications that handle personal transactions, business-to-business transactions, process credit card information, or process personally identifiable information. Level 2 provides some confidence in the correct use of security controls and confidence that the security controls themselves are working correctly. There are two constituent components for Level 2, as depicted in the figure below.

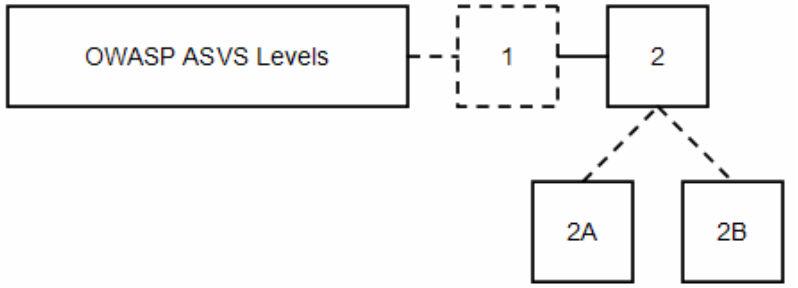

Figure 5 – OWASP ASVS Levels 2, 2A, and 2B

While it may be determined that an application meets either Level 2A or 2B, neither of these levels alone provide the same levels of rigor or coverage as Level 2. Further, while Level 2 is a superset of Level 1, there is no requirement to run an automated tool to meet the Level 2 requirements. Instead, the verifier has the option of using just manual techniques for all requirements. If automated tool results are available, the verifier may use them to support the analysis. However, even passing a requirement at Level 1 does not automatically indicate passing the same requirement at Level 2. This is because automated tools rely on signatures for problems, and do not provide sufficient evidence that the positive requirement has been met.

Manual techniques are still assumed to employ the use of tools. This can include the use of any kind of security analysis or testing tool, including the automated tools that are used for Level 1 verifications. However, such tools are simply aids to the analyst to find and assess the security controls being verified. Such tools may or may not contain logic to automatically detect application vulnerabilities. .

The following are minimal requirements for Level 2, 2A, or 2B web applications:

Security Control Behavior Requirements

L2.1: Verify that all security controls make decisions using a whitelist approach and that security controls cannot be bypassed according to the Level 2A and 2B requirements specified in the "Detailed Verification Requirements" section. (This is a new requirement at Level 2)

Security Control Use Requirements

None: There are no requirements for where web application security controls are used within the application at Level 2.

Security Control Implementation Requirements

None: There are no requirements for how web application security controls are built at Level 2.

Security Control Verification Requirements

L2.2: The verifier shall perform manual penetration testing on the web application according to the Level 2A requirements specified in the “Detailed Verification Requirements” section.

L2.3: The verifier shall perform manual source code review on the web application according to the Level 2B requirements specified in the “Detailed Verification Requirements” section.

Requirements that allow the use of either manual penetration testing or manual code review do not have to be verified with both. These verification requirements can be verified with either technique at Level 2.

The verifier may optionally perform automated source code or dynamic scanning on the web application as defined in Level 1. This automated verification cannot be used in place of the manual review of each requirement. However, if the scan results help the verifier perform their work more quickly, they can certainly be used to assist in performing a Level 2 verification. Note that because negative signatures cannot verify the proper design, implementation, or use of a security control, even a verified automated result for a requirement does not mean that the manual review required at Level 2 has been completed.

Documentation Requirements

L2.4: The verifier shall create a verification report that describes the web application security architecture, and the results of the verification according to the requirements in section “Verification Reporting Requirements”.

At Level 2, the web application shall be defined by grouping its components into a high-level architecture (for example MVC controller components, business function components, and data layer components). Components may be defined in terms of either individual or groups of source files, libraries, and/or executables. At Level 2, the relationship between components or groups of components need not be defined. For example, the diagram below depicts a web application that consists of a web server application, an application server application, custom code, libraries, and a database application that are grouped according to an MVC architecture. At Level 2, the path or paths a given end user request may take within the application must be documented, as depicted in the figure below. However, not all such paths must be examined.

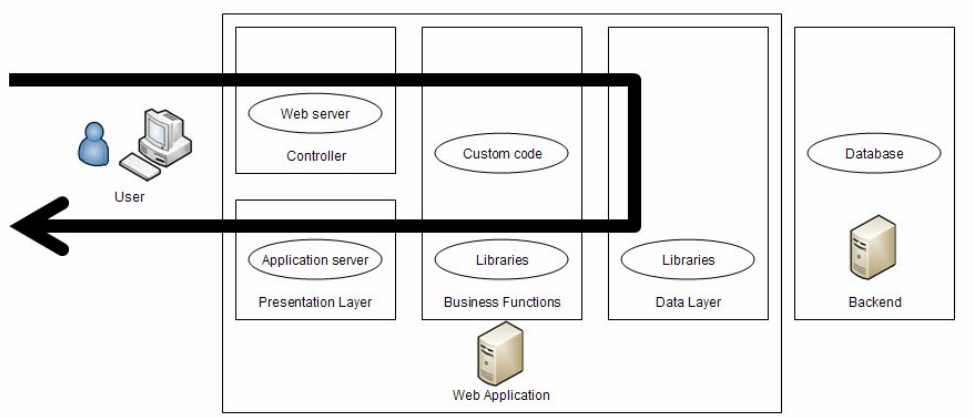

Figure 6 – OWASP ASVS Level 2 Security Architecture Example

LEVEL 2A – PENETRATION TEST (PARTIAL MANUAL VERIFICATION)

Manual Penetration Testing Security Control Verification Requirements

Manual penetration testing consists of creating dynamic tests to verify an application's proper design, implementation, and use of security controls.

L2A.1: The verifier shall perform manual penetration testing on the application according to the Level 2A requirements specified in the "Detailed Verification Requirements" section.

Where appropriate, the verifier may use sampling to establish the effective use of a security control. The verifier may choose to document a vulnerability pattern that will allow developers to confidently find and fix all instances of the pattern in the software baseline.

Multiple instances of a vulnerability pattern that can be traced to a single root cause should be combined into a single risk.

LEVEL 2B – CODE REVIEW (PARTIAL MANUAL VERIFICATION)

Manual Code Review Security Control Verification Requirements

Manual code review consists of human searching and analysis of web application source code to verify the web application's design, implementation, and use of security controls. Such analysis is expected to be tool assisted, but could simply involve commonly available tools such as a source code editor or IDE.

L2B.1: The verifier shall perform manual code review on the application according to the Level 2B requirements specified in the "Detailed Verification Requirements" section.

Where appropriate, the verifier may use sampling to establish the effective use of a security control. The verifier may choose to document a vulnerability pattern that

will allow developers to confidently find and fix all instances of the vulnerability pattern in the software baseline.

Multiple instances of a vulnerability pattern that can be traced to a single root cause should be combined into a single risk.

LEVEL 3 – DESIGN VERIFICATION

Level 3 ("Design Verification") is typically appropriate for applications that handle significant business-to-business transactions, including those that process healthcare information, implement business critical or sensitive functions, or process other sensitive assets.

Level 3 ensures that security controls themselves are working correctly, and that security controls are used everywhere within the application that they need to be used to enforce application-specific policies. Level 3 is not broken into constituent components, as depicted in the figure below.

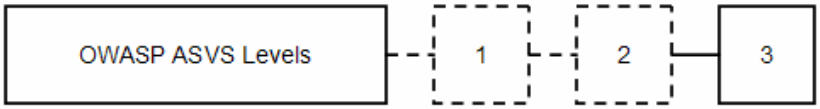

Figure 7 – OWASP ASVS Level 3

The following are minimal requirements for Level 3 web applications:

Security Control Behavior Requirements

L3.1: Verify that all security controls make decisions using a whitelist approach and that security controls cannot be bypassed according to the Level 3 requirements specified in the "Detailed Verification Requirements" section. (This requirement was introduced at Level 2)

Security Control Use Requirements

L3.2: Verify that all security controls are centralized within the web application, on the server side according to the Level 3 requirements specified in the "Detailed Verification Requirements" section. (This is a new requirement at Level 3)

Security Control Implementation Requirements

None: There are no requirements for how web application security controls are built at Level 3.

Security Control Verification Requirements

L3.3: The verifier shall perform manual verification of the web application according to the Level 2 requirements specified in the "Detailed Verification Requirements" section.

L3.4: In addition, the verifier shall create a security architecture and use it to verify the proper design and use of all security controls by performing threat modeling..

Documentation Requirements

L3.5: The verifier shall create a verification report that describes the web application security architecture, and the results of the verification according to the requirements specified in the “Verification Reporting Requirements” section.

The web application shall be defined by grouping its components into a high-level architecture (for example MVC controller components, business function components, and data layer components), and at Level 3, the relationship between components or groups of components must be defined. Components may be defined in terms of either individual or groups of source files, libraries, and/or executables. At Level 3, supporting threat modeling information about threat agents and assets must be provided. The path or paths a given end user request may take within the application must be documented, as depicted in the figure below. At Level 3, all paths through the application must be examined.

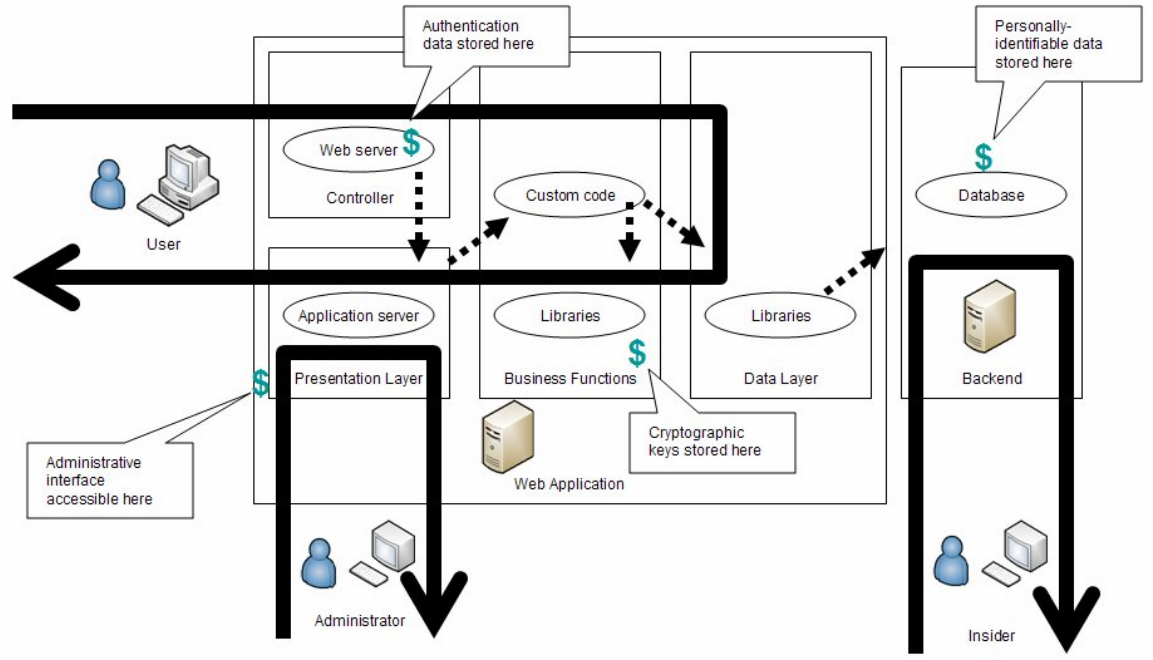

Figure 8 – OWASP ASVS Level 3 Security Architecture Example

LEVEL 4 – INTERNAL VERIFICATION

Level 4 (“Internal Verification”) is typically appropriate for critical applications that protect life and safety, critical infrastructure, or defense functions. Level 4 ensures that security controls themselves are working correctly, that security controls are used everywhere within the application that they need to be used to enforce application-specific policies, and that secure coding practices were followed. Level 4 is not broken out into constituent components, as depicted in the figure below.

Figure 9 – OWASP ASVS Level 4

The following are minimal requirements for Level 4 web applications:

Security Control Behavior Requirements

L4.1: Verify that all security controls make decisions using a whitelist ("positive") approach and that security controls cannot be bypassed according to the Level 4 requirements specified in the "Detailed Verification Requirements" section.

Security Control Use Requirements

L4.2: Verify that all security controls are centralized within the web application, on the server side according to the Level 4 requirements specified in the "Detailed Verification Requirements" section.

Security Control Implementation Requirements

L4.3: Verify that the web application does not contain any malicious code according to the Level 4 requirements specified in the "Detailed Verification Requirements" section. (This is a new requirement at Level 4)

Security Control Verification Requirements

L4.4: A prerequisite for Level 4 is that the verifier shall create a security architecture and verify the security controls for the web application by performing threat modeling according to the Level 3 requirements specified in the "Detailed Verification Requirements" section.

L4.5: The verifier shall perform a manual review of the entire code base to search for malicious code (which is not the same as malware) according to the Level 4 requirements specified in the "Detailed Verification Requirements" section. (This is a new requirement at Level 4)

Documentation Requirements

L4.6: The verifier shall create a verification report that describes the web application security architecture, and the results of the verification according to the requirements specified in the "Verification Reporting Requirements" section.

The web application architecture shall be captured as required at Level 3. Further, Level 4 requires that all application code, including code not explicitly examined, be identified as part of the web application definition, as depicted in the figure below. This code must include all libraries, frameworks, and supporting code that the application relies on. Previous verifications of these components can be reused as part of another verification.

Platform code, such as the operating system, virtual machine, or core libraries issued with a virtual machine environment, web server, or application server are not included in Level 4. For example, libraries associated with the Java runtime would not need to be assessed at Level 4.

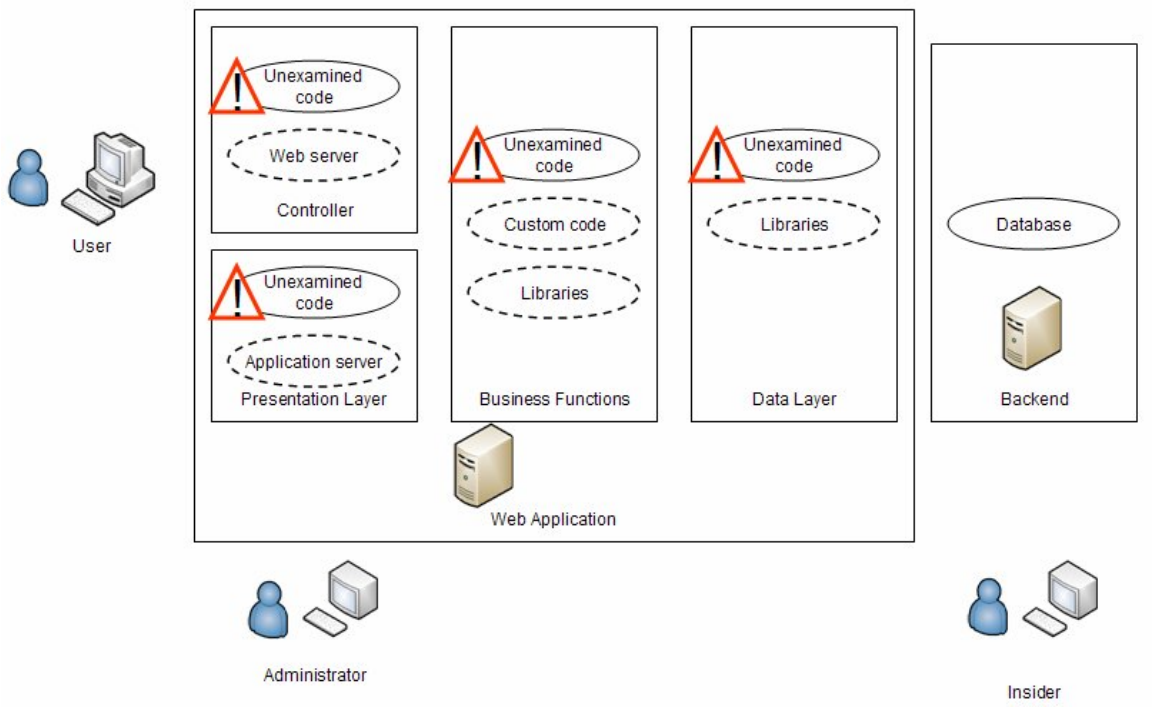

Figure 10 – OWASP ASVS Level 4 Unexamined Code Example

SOME GUIDANCE ON THE VERIFICATION PROCESS

Verifying an application against the detailed verification requirements can be a complex task. Here is some guidance for certain situations.

1) Automated Verifications

As stated previously in the section describing LEVEL 1 – AUTOMATED VERIFICATION:

> "Because automated tools generally use vulnerability signatures to find problems, this level only provides partial web application security verification coverage."

This means that automated tools are generally signature based, where those signatures look for patterns of bad behavior which indicate there is a likely flaw in the application. Such tools do not usually perform a positive analysis on the application to verify that it is correct throughout the application with regard to the particular vulnerability being looked for. For example, tools may be good at finding certain types of cross site scripting (XSS) vulnerabilities, but may not be good at finding all types, or may have difficulty when certain technologies are used, like AJAX. We don't expect automated tools to be able detect all flaws in any of these areas.

For Level 1, if an application passes the signatures, then it passes. The verifier simply needs to explain what the signatures actually do in the report so the reader can understand the benefit of the tool(s) used for each particular verification requirement.

2) Assessment of Off-The-Shelf Components as part of Automated or Manual Code Review (LEVELS 1B & 2B)

If you are performing a code review and you do not have access to the code for the security control being used, then you need to review the control's documentation and (for Level 2B) any security configuration in order to verify that the security control meets the specified requirement. Similarly, if you do have access to the code, but the security control is data driven, you also have to review the configuration data (for Level 2B) to make sure the control is configured properly for this application, not just that the security mechanism works generically.

The results of your documentation analysis and any configuration data review should be documented in the verification report.

3) Assessment of generic applications

If you are performing a review of a generic instance of an application rather than an application in a specific target environment, then the verification effort needs to make sure that the provided security controls work as advertised. This means determining what the provided security controls are supposed to be capable of, and then verifying that they work correctly in a variety of common configurations.

If the generic application does not provide or only partially provides security controls in certain areas, and expects the deployment environment to provide them, then the verification report should state for each affected verification requirement what the application does or does not provide. The report should also state whether the current state of practice makes it reasonable to expect a deployment environment to provide the expected capabilities. It is expected that only a few ASVS requirements could be completely met by externally provided controls.

DETAILED VERIFICATION REQUIREMENTS

This section of the OWASP Application Security Verification Standard (ASVS) defines derived verification requirements that apply for each of the verification levels this standard defines. Each section below defines a set of detailed verification requirements grouped into related areas. The ASVS defines the following verification requirements areas:

V1. Security Architecture
V2. Access Control
V3. Authentication
V4. Session Management
V5. Input Validation
V6. Output Encoding/Escaping
V7. Cryptography
V8. Error Handling and Logging
V9. Data Protection
V10. Communication Security
V11. HTTP Security

For each of these areas, the requirements that must be met at each of the verification levels listed below will be specified:

Level 1: Automated Verification
- Level 1A – Dynamic Scan (Partial Automated Verification)
- Level 1B – Source Code Scan (Partial Automated Verification)

Level 2: Manual Verification
- Level 2A – Penetration Test (Partial Manual Verification)
- Level 2B – Code Review (Partial Manual Verification)

Level 3: Design Verification
Level 4: Internal Verification

V1 – SECURITY ARCHITECTURE VERIFICATION REQUIREMENTS

For all levels, defining a security architecture is necessary to ensure both the completeness and accuracy (and repeatability when remediation is required) of the application security verification that is performed. Analysis can be directed and results can be traced back to an application's security architecture. However, while a security architecture must be defined at each verification level, the information provided and level of detail necessary is different, depending on the level.

The table below defines the corresponding security architecture verification requirements that apply for each of the four verification levels.

Table 1 – OWASP ASVS Security Architecture Requirements (V1)

Verification Requirement	Level 1A	Level 1B	Level 2A	Level 2B	Level 3	Level 4
V1.1: The verifier shall identify all application components (either individual or groups of source files, libraries, and/or executables) present in the application.	✓	✓	✓	✓	✓	✓
V1.2: The verifier shall define or verify the existence of a high-level architecture for the application.			✓	✓	✓	✓
V1.3: The verifier shall identify the logical interconnections between application components, groups of components, and external systems.					✓	✓
V1.4: Verify that the security architecture for the application has been documented.					✓	✓
V1.5: The verifier shall verify the integrity of interpreted code, libraries, executables, and configuration files using checksums or hashes.						✓

V2 – AUTHENTICATION VERIFICATION REQUIREMENTS

The Authentication Verification Requirements define a set of requirements for generating and handling account credentials safely.

The table below defines the corresponding verification requirements that apply for each of the four verification levels.

Table 2 – OWASP ASVS Authentication Requirements (V2)

Verification Requirement	Level 1A	Level 1B	Level 2A	Level 2B	Level 3	Level 4
V2.1: Verify that all pages and resources require authentication except those specifically intended to be public.	✓	✓	✓	✓	✓	✓
V2.2: Verify that for each authentication control (e.g., username/password authentication control; certificate-based authentication control), there is a single implementation that is used by the application.				✓	✓	✓
V2.3: Verify that all authentication controls are enforced on the server side.			✓	✓	✓	✓
V2.4: Verify that all code implementing or using authentication controls is not affected by any malicious code.						✓
V2.5: Verify that all authentication controls fail securely.			✓	✓	✓	✓
V2.6: Verify that all authentication decisions are logged.				✓	✓	✓
V2.7: Verify that the strength of any authentication credentials are sufficient to withstand brute force attacks possible in the deployed environment.			✓	✓	✓	✓
V2.8: Verify that account passwords are salted and hashed before storing.				✓	✓	✓
V2.9: Verify that if a maximum number of authentication attempts is exceeded, the account is locked.	✓		✓	✓	✓	✓
V2.10: Verify that all account management functions are at least as resistant to attack as the primary authentication mechanism.			✓	✓	✓	✓
V2.11: Verify that all password fields do not echo the user's password when it is entered.	✓	✓	✓	✓	✓	✓

V2.12: Verify that users can safely change their credentials.			✓	✓	✓	✓
V2.13: Verify that all authentication credentials are encrypted and stored in a centralized location (not in source code).				✓	✓	✓
V2.14: Verify that re-authentication is required before any application-specific sensitive operations are permitted.			✓	✓	✓	✓

V3 – SESSION MANAGEMENT VERIFICATION REQUIREMENTS

The Session Management Verification Requirements define a set of requirements for safely using HTTP requests, responses, sessions, cookies, headers, and logging to manage sessions properly.

The table below defines the corresponding verification requirements that apply for each of the four verification levels.

Table 3 – OWASP ASVS Session Management Requirements (V3)

Verification Requirement	Level 1A	Level 1B	Level 2A	Level 2B	Level 3	Level 4
V3.1: Verify that the framework's default session management control implementation is used by the application.	✓		✓	✓	✓	✓
V3.2: Verify that all code implementing or using session management controls is not affected by any malicious code.						✓
V3.3: Verify that sessions are invalidated when the user logs out.	✓		✓	✓	✓	✓
V3.4: Verify that sessions timeout after a specified period of inactivity.	✓		✓	✓	✓	✓
V3.5: Verify that sessions timeout after a maximum time period regardless of activity (an absolute timeout).					✓	✓
V3.6: Verify that the session id is changed on login, reauthentication, and logout.			✓	✓	✓	✓

V3.7: Verify that the session id is never disclosed other than in cookie headers, particularly in URLs or logs. This includes verifying that the application does not support URL rewriting of session cookies.		✓		✓	✓	✓
V3.8: Verify that all authenticated pages have logout links.	✓		✓	✓	✓	✓
V3.9: Verify that only session ids generated by the application framework are recognized as valid by the application.			✓	✓	✓	✓

V4 – ACCESS CONTROL VERIFICATION REQUIREMENTS

The Access Control Verification Requirements define how an application can safely enforce access control. In most applications, access control must be performed in multiple different locations across the various application layers. These requirements define verification requirements for access controls for URLs, business functions, data, services, and files.

The table below defines the corresponding verification requirements that apply for each of the four verification levels.

Table 4 – OWASP ASVS Access Control Requirements (V4)

Verification Requirement	Level 1A	Level 1B	Level 2A	Level 2B	Level 3	Level 4
V4.1: Verify that users can only access URLs for which they possess specific authorization.	✓		✓	✓	✓	✓
V4.2: Verify that users can only access files for which they possess specific authorization.	✓		✓	✓	✓	✓
V4.3: Verify that directory browsing is disabled unless deliberately desired.	✓		✓		✓	✓
V4.4: Verify that users can only access protected functions for which they possess specific authorization.	✓	✓	✓	✓	✓	✓
V4.5: Verify that users can only access services for which they possess specific authorization.			✓	✓	✓	✓

V4.6: Verify that users can only access data for which they possess specific authorization.			✓	✓	✓	✓
V4.7: Verify that access controls fail securely.			✓	✓	✓	✓
V4.8: Verify that access control decisions can be logged and all failed decisions are logged.				✓	✓	✓
V4.9: Verify that the same access control rules implied by the presentation layer are enforced on the server side.			✓	✓	✓	✓
V4.10: Verify that all information used by access controls cannot be manipulated by end users unless specifically authorized.			✓	✓	✓	✓
V4.11: Verify that direct object references are protected, such that only authorized objects are accessible to each user.	✓		✓	✓	✓	✓
V4.12: Verify that for each type of resource protected, there is a single security control that is used to protect access to that type of resource.				✓	✓	✓
V4.13: Verify that all access controls are enforced on the server side.			✓	✓	✓	✓
V4.14: Verify that all code implementing or using access controls is not affected by any malicious code.						✓
V4.15: Verify that limitations on input and access imposed by the business on the application (such as daily transaction limits) cannot be bypassed.			✓	✓	✓	✓

V5 – INPUT VALIDATION VERIFICATION REQUIREMENTS

The Input Validation Requirements define a set of requirements for validating input so that it is safe for use within an application.

The table below defines the corresponding verification requirements that apply for each of the four verification levels.

Table 5 – OWASP ASVS Input Validation Requirements (V5)

Verification Requirement	Level 1A	Level 1B	Level 2A	Level 2B	Level 3	Level 4
V5.1: Verify that a character set, such as UTF-8, is specified for all sources of input.			✓	✓	✓	✓
V5.2: Verify that all input data is canonicalized for all downstream decoders or interpreters prior to validation.					✓	✓
V5.3: Verify that a positive validation pattern is defined and applied to all input.	✓		✓	✓	✓	✓
V5.4: Verify that a single input validation control is used by the application.				✓	✓	✓
V5.5: Verify that all input validation is performed on the server side.			✓	✓	✓	✓
V5.6: Verify that all input validation controls are not affected by any malicious code.						✓
V5.7: Verify that all input validation control failures result in input rejection.	✓		✓	✓	✓	✓
V5.8: Verify that all input validation failures are logged.				✓	✓	✓
V5.9: Verify that the environment is not susceptible to buffer overflows, or that security controls prevent buffer overflows.	✓	✓	✓	✓	✓	✓

V6 – OUTPUT ENCODING/ESCAPING VERIFICATION REQUIREMENTS

The Output Encoding/Escaping Validation Requirements define a set of requirements for verifying that output is properly encoded so that it is safe for external applications.

The table below defines the corresponding verification requirements that apply for each of the four verification levels.

Table 6 – OWASP ASVS Output Encoding/Escaping Requirements (V6)

Verification Requirement	Level 1A	Level 1B	Level 2A	Level 2B	Level 3	Level 4
V6.1: Verify that output encoding / escaping controls encode all characters not known to be safe for the intended interpreter.				✓	✓	✓
V6.2: Verify that for each type of output encoding/escaping performed by the application, there is a single security control for that type of output.					✓	✓
V6.3: Verify that all output encoding/escaping controls are implemented on the server side.			✓	✓	✓	✓
V6.4: Verify that all code implementing or using output validation controls is not affected by any malicious code.						✓
V6.5: Verify that all untrusted data that are output to HTML (including HTML elements, HTML attributes, javascript blocks, javascript event handlers, CSS blocks, and URI attributes) are properly escaped for the applicable context.		✓	✓	✓	✓	✓
V6.6: Verify that all untrusted data that are output to SQL interpreters use parameterized interfaces, prepared statements, or are escaped properly.				✓	✓	✓
V6.7: Verify that all untrusted data that are output to XML use parameterized interfaces or are escaped properly.				✓	✓	✓
V6.8: Verify that all untrusted data that are used in LDAP queries are escaped properly.				✓	✓	✓
V6.9: Verify that all untrusted data that are included in operating system command parameters are escaped properly.				✓	✓	✓

V6.10: Verify that all untrusted data that are output to any interpreters not specifically listed above are escaped properly.				✓	✓	✓

V7 – CRYPTOGRAPHY VERIFICATION REQUIREMENTS

The Encryption Verification Requirements define a set of requirements that can be used to verify a web application's encryption, key management, random number, and hashing operations. Web applications should always use FIPS 140-2 validated cryptographic modules, or cryptographic modules validated against an equivalent standard (e.g., a non-U.S. standard).

The table below defines the corresponding verification requirements that apply for each of the four verification levels.

Table 7 – OWASP ASVS Cryptography Requirements (V7)

Verification Requirement	Level 1A	Level 1B	Level 2A	Level 2B	Level 3	Level 4
V7.1: Verify that cryptographic modules used by the application have been validated against FIPS 140-2 or an equivalent standard. (See http://csrc.nist.gov/groups/STM/cmvp/validation.html).				✓	✓	✓
V7.2: Verify that cryptographic modules operate in their approved mode.			✓	✓	✓	✓
V7.3: Verify that access to the master secret(s) is protected from unauthorized access (A master secret is an application credential stored as plaintext on disk that is used to access security configuration information).				✓	✓	✓
V7.4: Verify that password hashes are salted when they are created.				✓	✓	✓
V7.5: Verify that all cryptographic operations are performed server side.			✓	✓	✓	✓
V7.6: Verify that all code using the cryptographic module is not affected by any malicious code.						✓
V7.7: Verify that cryptographic modules fail securely.				✓	✓	✓

V7.8: Verify that cryptographic module failures are logged.				✓	✓	✓
V7.9: Verify that all random numbers, random file names, random GUIDs, and random strings are generated using the cryptographic module's approved random number generator for instances when not using random numbers to avoid collisions.				✓	✓	✓
V7.10: Verify that the application supports a key change mechanism that is appropriate for the given use of each cryptographic module.					✓	✓

V8 – ERROR HANDLING AND LOGGING VERIFICATION REQUIREMENTS

The Error Handling and Logging Verification Requirements define a set of requirements that can be used to verify the tracking of security relevant events and the identification of attack behavior.

The table below defines the corresponding verification requirements that apply for each of the four verification levels.

Table 8 – OWASP ASVS Error Handling and Logging Requirements (V8)

Verification Requirement	Level 1A	Level 1B	Level 2A	Level 2B	Level 3	Level 4
V8.1: Verify security logging controls provide the ability to log both success and failure events that are identified as security-relevant.				✓	✓	✓

V8.2: Verify that each log event includes: 1. a time stamp from a reliable source, 2. severity level of the event, 3. an indication that this is a security relevant event (if mixed with other logs) 4. the identity of the user that caused the event, 5. the source IP address of the request associated with the event 6. whether the event succeeded or failed, and 7. a description of the event				✓	✓	✓
V8.3: Verify that security logs are protected from unauthorized access.				✓	✓	✓
V8.4: Verify that there is a single logging implementation that is used by the application.				✓	✓	✓
V8.5: Verify that all error handling and logging controls are performed on the server side.			✓	✓	✓	✓
V8.6: Verify that all code implementing or using error handling and logging controls is not affected by any malicious code.						✓
V8.7: Verify that that the application does not log sensitive data that could assist an attacker, including session id and personal or sensitive information.				✓	✓	✓
V8.8: Verify that that the application does not output error messages containing sensitive data that could assist an attacker, including session id and personal information.	✓	✓	✓	✓	✓	✓
V8.9: Verify that the security logging system includes a log analysis tool which allows the analyst to search for log events based on combinations of search criteria across all fields in the log record format supported by this system.				✓	✓	✓

V9 – DATA PROTECTION VERIFICATION REQUIREMENTS

The Data Protection Verification Requirements define a set of requirements that can be used to verify the protection of sensitive data (e.g. credit card number, SSN, privacy data).

The table below defines the corresponding verification requirements that apply for each of the four verification levels.

Table 9 – OWASP ASVS Data Protection Security Requirements (V9)

Verification Requirement	Level 1A	Level 1B	Level 2A	Level 2B	Level 3	Level 4
V9.1: Verify that the list of sensitive data processed by this application is identified, and that there is an explicit policy for how access to this data must be controlled, and when this data must be encrypted (both at rest and in transit).				✓	✓	✓
V9.2: Verify that all cached or temporary copies of sensitive data (both server side and client side) are protected from unauthorized access or purged/invalidated after the authorized user accesses the sensitive data.				✓	✓	✓
V9.3: Verify that all forms containing sensitive information have disabled client side caching, including autocomplete features.	✓	✓	✓	✓	✓	✓
V9.4: Verify that all sensitive data is sent to the server in the HTTP message body (i.e., URL parameters are never used to send sensitive data).			✓		✓	✓

V10 – COMMUNICATION SECURITY VERIFICATION REQUIREMENTS

The Communication Security Verification Requirements define a set of requirements that can be used to verify that all communications with web application are properly secured.

The table below defines the corresponding verification requirements that apply for each of the four verification levels.

Table 10 – OWASP ASVS Communication Security Requirements (V10)

Verification Requirement	Level 1A	Level 1B	Level 2A	Level 2B	Level 3	Level 4
V10.1: Verify that TLS is used for all connections (including both external and backend connections) that are authenticated or that involve sensitive data or functions.				✓	✓	✓
V10.2: Verify that there is a single standard TLS implementation that is used by the application that is configured to operate in an approved mode of operation (See http://csrc.nist.gov/groups/STM/cmvp/documents/fips140-2/FIPS1402IG.pdf).					✓	✓
V10.3: Verify that failed TLS connections do not fall back to an insecure connection.			✓		✓	✓
V10.4: Verify that backend TLS connection failures are logged.				✓	✓	✓
V10.5: Verify that TLS server certificates have been issued by a trusted CA.	✓		✓	✓	✓	✓
V10.6: Verify that certificate paths are built and verified for all client certificates using configured trust anchors and revocation information.				✓	✓	✓
V10.7: Verify that specific character encodings are defined for all connections (e.g., UTF-8).			✓	✓	✓	✓
V10.8: Verify that all connections to external systems that involve sensitive information or functions are authenticated and use a minimally privileged account.				✓	✓	✓

V11 – HTTP SECURITY VERIFICATION REQUIREMENTS

The HTTP Security Verification Requirements define a set of requirements that can be used to verify security related to HTTP requests, responses, sessions, cookies, headers, and logging.

The table below defines the corresponding verification requirements that apply for each of the four verification levels.

Table 11 – OWASP ASVS HTTP Security Requirements (V11)

Verification Requirement	Level 1A	Level 1B	Level 2A	Level 2B	Level 3	Level 4
V11.1: Verify that the HTTPOnly flag is used on all cookies that do not specifically require access from JavaScript.			✓	✓	✓	✓
V11.2: Verify that the secure flag is used on all cookies that contain sensitive data, including the session cookie.			✓	✓	✓	✓
V11.3: Verify that the application generates a random token as part of all links and forms associated with transactions or accessing sensitive data, and that the application verifies the presence of the token in corresponding requests.					✓	✓
V11.4: Verify that every HTTP response contains a content type header specifying a safe character set (e.g., UTF-8).	✓	✓	✓	✓	✓	✓
V11.5: Verify that redirects do not include unvalidated data.	✓	✓	✓	✓	✓	✓
V11.6: Verify that the application accepts only a defined set of HTTP request methods, such as GET and POST.	✓	✓	✓	✓	✓	✓
V11.7: Verify that HTTP headers in both requests and responses contain only printable ASCII characters.			✓	✓	✓	✓

V12 – SECURITY CONFIGURATION VERIFICATION REQUIREMENTS

The Security Configuration Verification Requirements define a set of requirements that can be used to verify the secure storage of all configuration information that directs the security-related behavior of the web application. Protection of this configuration information is critical to the secure operation of the application.

The table below defines the corresponding verification requirements that apply for each of the four verification levels.

Table 12 – OWASP ASVS Security Configuration Requirements (V12)

Verification Requirement	Level 1A	Level 1B	Level 2A	Level 2B	Level 3	Level 4
V12.1: Verify that all security-relevant configuration information is stored in locations that are protected from unauthorized access.				✓	✓	✓
V12.2: Verify that all changes to the security configuration are logged in the security event log.					✓	✓
V12.3: Verify that all access to the application is denied if the application cannot access its security configuration information.				✓	✓	✓
V12.4: Verify that the configuration store can be output in a human-readable format to facilitate audit.						✓

V13 – MALICIOUS CODE SEARCH VERIFICATION REQUIREMENTS

For Level 4, searching for malicious code in any code that has not yet been examined after performing a Level 3 application verification is required.

The table below defines the Malicious Code Search requirements that are introduced at Level 4.

Table 13 – OWASP ASVS Malicious Code Search Requirements (V13)

Verification Requirement	Level 1A	Level 1B	Level 2A	Level 2B	Level 3	Level 4
V13.1: Verify that there are no time bombs in any unexamined code by examining all system clock calls.						✓
V13.2: Verify that there are no back doors in any unexamined code by examining all code for functions unrelated to business requirements.						✓
V13.3: Verify that there are no Easter eggs in any unexamined code by examining all execution paths for extraneous code.						✓

V13.4: Verify that there are no salami attacks in any unexamined code by examining all financial transactions for incorrect logic.						✓
V13.5: Verify that there are no other types of malicious code in any unexamined code.						✓

V14 – INTERNAL SECURITY VERIFICATION REQUIREMENTS

For Level 4, searching for malicious code in code that has been examined after performing Level 3 application verification is required.

The table below defines the corresponding verification requirements that apply for each of the four verification levels.

Table 4 – OWASP ASVS Self-Protection Requirements (V14)

Verification Requirement	Level 1A	Level 1B	Level 2A	Level 2B	Level 3	Level 4
V14.1: Verify that the application protects all information used by security controls from misuse.					✓	✓
V14.2: Verify that security control interfaces are simple enough to use that developers are likely to use them correctly.						✓
V14.3: Verify that the application properly protects shared variables and resources from inappropriate concurrent access.						✓

VERIFICATION REPORTING REQUIREMENTS

An OWASP Application Security Verification Standard Report contains a description of the web application that was analyzed against the OWASP Application Security Verification Standard. The Report also documents the results of the analysis, including any remediation of vulnerabilities that was required.

An OWASP Application Security Verification Standard Report shall conform to the content requirements described in this section. A Report should include all necessary material necessary for a reader to understand the analysis that was performed and the results of the analysis, including configuration information and code snippets, as appropriate.

The contents of an OWASP Application Security Verification Standard Report are depicted in the figure below, which should be used when constructing the Report outline.

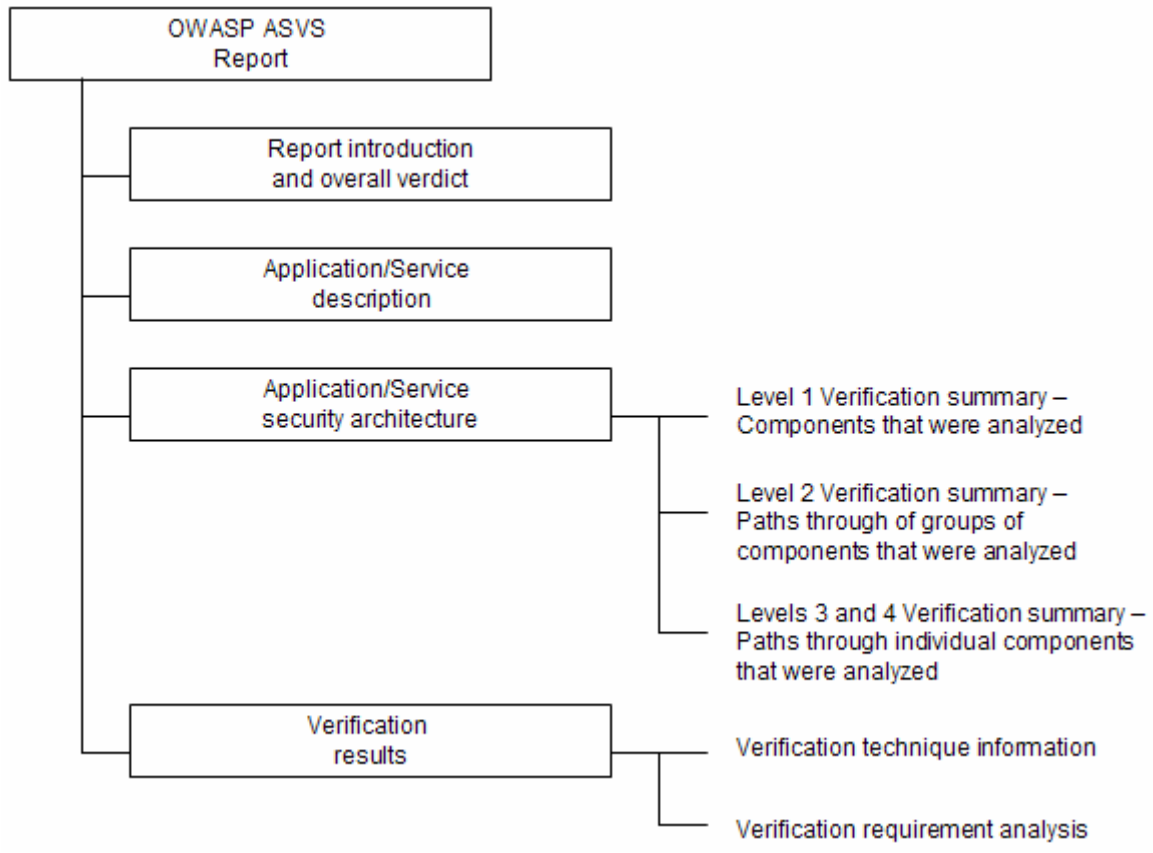

Figure 11 – OWASP ASVS Report Contents

R1 – REPORT INTRODUCTION

This part of the Report shall provide sufficient information to identify both the Report and the web application that is the subject of the report. The Report introduction shall also summarize the overall verdict.

R2 – APPLICATION DESCRIPTION

This part of the Report shall provide sufficient description of the web application to aid the understanding of its operation and the environment that it operates in.

R3 – APPLICATION SECURITY ARCHITECTURE

This part of the Report shall provide additional detail describing the web application as the first step in providing confidence to the reader of the report that the analysis that was performed was both complete and accurate. This part of the Report provides context for the analysis. The information presented in this section will be used in the course of the analysis to identify inconsistencies.

This part of the Report shall provide different levels of detail, depending on the OWASP Application Security Verification Standard Level that the analysis was performed. Details will vary according to Level as follows:

- Level 1 Verification – This part of the Report shall identify and describe components that were analyzed as defined in the Verification Level 1 section of the ASVS.
- Level 2 Verification Summary – This part of the Report shall identify and describe paths through of groups of components that were analyzed as defined in the Verification Level 2 section of the ASVS.
- Levels 3 Verification Summary – This part of the Report shall identify and describe paths through individual components that were analyzed as defined in the Verification Level 3 section of the ASVS.
- Levels 4 Verification Summary – This part of the Report shall identify and describe code analyzed as defined in the Verification Level 2 section of the ASVS.

R4 – VERIFICATION RESULTS

This part of the Report shall provide the results of the analysis that was performed according to section “Verification Requirements” of the Standard, including description of any remediation of vulnerabilities that was required, as follows:

Table 14 – OWASP ASVS Report Verification Results Contents

Level	Pass	Fail
Level 1 Automated Verification	• Verdict • Location (URL and/or source file path, name and line number) • Verdict justification (description of scan configuration)	• Verdict • Location (URL and/or source file path, name and line number) • Description (including configuration information as appropriate) • Risk rating (see the OWASP Risk Rating Methodology) • Risk justification
Level 2 Manual Implementation Verification	• Verdict • Location (URL and/or source file path, name and line number) • Verdict justification (an argument for completeness and correctness, providing specific evidence)	• Verdict • Location (URL and/or source file path, name and line number) • Description (including path through application components and steps to reproduce) • Risk rating (see the OWASP Risk Rating Methodology) • Risk justification

Level 3 Design Verification	• Verdict • Affected components and any relevant locations (URL and/or source file path, name and line number) • Verdict justification (an argument for completeness and correctness, providing specific evidence)	• Verdict • Affected components and any relevant locations (URL and/or source file path, name and line number) • Description (including path through application components and steps to reproduce) • Risk rating (see the OWASP Risk Rating Methodology) • Risk justification
Level 4 Internal Security Verification	• Verdict • Affected components and any relevant locations (URL and/or source file path, name and line number) • Verdict justification (an argument for completeness and correctness, providing specific evidence)	• Verdict • Affected components and any relevant locations (URL and/or source file path, name and line number) • Description (including path through application components and steps to reproduce) • Risk rating (see the OWASP Risk Rating Methodology) • Risk justification

GLOSSARY

Access Control – A means of restricting access to files, referenced functions, URLs, and data based on the identity of users and/or groups to which they belong.

Application Component – An individual or group of source files, libraries, and/or executables, as defined by the verifier for a particular web application.

Application Security Verification – The technical assessment of a web application against the OWASP ASVS.

Application Security Verification Report – A report that documents the overall results and supporting analysis produced by the verifier for a particular application.

Application Security Verification Standard (ASVS) – An OWASP standard that defines four levels of application security verification for web applications.

Authentication – The verification of the claimed identity of a web application user.

Automated Verification – The use of automated tools (either dynamic analysis tools, static analysis tools, or both) that use vulnerability signatures to find problems.

Back Doors – A type of malicious code that allows unauthorized access to a web application.

Common Criteria (CC) – A multipart standard that can be used as the basis for the verification of the design and implementation of security controls in IT products.

Communication Security – The protection of web application data when it is transmitted between application components, between web browsers and web applications, and between external systems and web applications.

Design Verification – The technical assessment of the security architecture of a web application.

Internal Verification – The technical assessment of specific aspects of the security architecture of a web application as defined in the OWASP ASVS.

Cryptographic module – Hardware, software, and/or firmware that implements cryptographic algorithms and/or generates cryptographic keys.

Denial of Service (DOS) Attacks – The flooding of a web application with more requests than it can handle.

Dynamic Verification – The use of automated tools that use vulnerability signatures to find problems in running web applications.

Easter Eggs – A type of malicious code that does not run until a specific user input event occurs.

External Systems – A server-side application or service that is not part of the web application.

FIPS 140-2 – A standard that can be used as the basis for the verification of the design and implementation of cryptographic modules

Input Validation – The canonicalization and validation of untrusted user input.

Malicious Code – Source code that is introduced into a web application during its development without the knowledge of the web application owner. Not the same as malware!

Malware – Executable code that is introduced into a web application during runtime without the knowledge of the web application user or administrator.

Open Web Application Security Project (OWASP) – The Open Web Application Security Project (OWASP) is a worldwide free and open community focused on improving the security of application software. Our mission is to make application security "visible," so that people and organizations can make informed decisions about application security risks. See: http://www.owasp.org

Output Validation – The canonicalization and validation of web application output to web browsers and to external systems.

OWASP Enterprise Security API (ESAPI) – A free and open collection of all the security methods that a developer needs to build a secure web application. See: http://www.owasp.org/index.php/ESAPI

OWASP Risk Rating Methodology – A risk rating methodology that has been customized for application security. See: http://www.owasp.org/index.php/How_to_value_the_real_risk

OWASP Testing Guide – A document designed to help organizations understand what comprises a testing program, and to help them identify the steps needed to build and operate that testing program.

OWASP Top Ten – A document that represents a broad consensus about what the most critical web application security flaws are.

Salami Attacks – A type of malicious code that is used to redirect small amounts of money without detection in financial transactions.

Security Architecture – An abstraction of a web application's design that identifies and describes where and how security controls are used, and also identifies and describes the location and sensitivity of both user and application data.

Security Configuration – The runtime configuration of a by a web application that affects how security controls are used.

Static Verification – The use of automated tools that use vulnerability signatures to find problems in web application source code.

Threat Modeling - A technique consisting of developing increasingly refined security architectures to identify threat agents, security zones, security controls, and important technical and business assets.

Time Bombs – A type of malicious code that does not run until a preconfigured time or date elapses.

Verifier - The person or team that is reviewing the web application against the OWASP ASVS requirements.

WHERE TO GO FROM HERE

OWASP is the premier site for web application security. The OWASP site hosts many projects, forums, blogs, presentations, tools, and papers. OWASP hosts two major web application security conferences per year, and has over 80 local chapters.

The following OWASP projects are most likely to be useful:

- OWASP Application Security Verification Standard Project, http://www.owasp.org/index.php/ASVS
- OWASP Top Ten Project, http://www.owasp.org/index.php/Top_10
- OWASP Enterprise Security API (ESAPI) Project, http://www.owasp.org/index.php/ESAPI
- OWASP Risk Rating Methodology, http://www.owasp.org/index.php/How_to_value_the_real_risk
- OWASP Testing Guide, http://www.owasp.org/index.php/Testing_Guide

Similarly, the following web sites are most likely to be useful:

- OWASP, http://www.owasp.org
- MITRE, Common Weakness Enumeration – Vulnerability Trends, http://cwe.mitre.org/documents/vuln-trends.html
- PCI Security Standards Council, publishers of the PCI standards, relevant to all organizations processing or holding credit card data, https://www.pcisecuritystandards.org
- PCI Data Security Standard (DSS) v1.1, https://www.pcisecuritystandards.org/pdfs/pci_dss_v1-1.pdf

THE BELOW ICONS REPRESENT WHAT OTHER VERSIONS ARE AVAILABLE IN PRINT FOR THIS TITLE BOOK.

ALPHA: "Alpha Quality" book content is a working draft. Content is very rough and in development until the next level of publication.

BETA: "Beta Quality" book content is the next highest level. Content is still in development until the next publishing.

RELEASE: "Release Quality" book content is the highest level of quality in a books title's lifecycle, and is a final product.

ALPHA
PUBLISHED

BETA
PUBLISHED

RELEASE
PUBLISHED

YOU ARE FREE:

to share - to copy, distribute and transmit the work

to Remix - to adapt the work

UNDER THE FOLLOWING CONDITIONS:

Attribution. You must astribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).

Share Alike. - If you alter, transform, or build upon this work, you may distribute the resulting work only under the same, similar or a compatible license.

OWASP
The Open Web Application Security Project

The Open Web Application Security Project (OWASP) is a worldwide free and open community focused on improving the security of application software. Our mission is to make application security "visible," so that people and organizations can make informed decisions about application security risks. Everyone is free to participate in OWASP and all of our materials are available under a free and open software license. The OWASP Foundation is a 501c3 not-for-profit charitable organization that ensures the ongoing availability and support for our work.

On the cover: Braconid wasps are beneficial parasites. Braconids parasitize a broad range of hosts: caterpillars, flies, wasps, beetles, and aphids. After a female injects an egg into a host, the larva feeds slowly on that single host. By the time the host dies, the larva is fully grown. It pupates inside or near the dead host, sometimes in a silken cocoon, to emerge later as an adult wasp.