SERVICE DESCRIPTION

Application Source Code Review

Service Scope

Trustwave's SpiderLabs Application Source Code review combines Static Analysis Security Testing (SAST) techniques with manual review and testing techniques of the target application, providing a deliverable with both tactical and strategic recommendations to improve the security posture of such target application. These recommendations are both actionable and advisory in nature and are presented to the customer.

The process involves methodical and expert driven testing of the target application to determine if the application is vulnerable to application layer security risks. This level of testing validates the application layer security controls and the security effectiveness of software development and deployment standards by determining how resilient the web application is to determined attackers.

During the Trustwave Application Source Code Review, SpiderLabs manually inspect relevant application source code to:

- Pinpoint deficiencies in security controls
- Identify development errors that violate best-practices
- Identify development errors that lead to vulnerabilities
- Evaluate the third-party tools, applications, and libraries used to create and run the front and back-end services

Trustwave SpiderLabs conducts application code reviews for Clients to assess and improve the security of their applications. During the course of this review, SpiderLabs conducts a detailed inspection of the source code of the application, and assess any vulnerabilities in the third-party tools, applications, and libraries used to create and run the front and back-end application services.

Below is a non-exhaustive list of the typical areas of focus for an application source code review that can be tailored upon request in conjunction with Client:

- Input Handling, Data Validation and Injection Flaws
- Data type conversions
- Authentication and Session Management
- Direct Object References
- Cryptographic Methods
- Cryptographic Storage

Copyright © 2018 Trustwave Holdings, Inc. All rights reserved.

- Back-end database queries
- Use of dynamic arguments
- Review of server-side components (servlets)
- Database, RPC calls and interfaces to outside systems

Deliverables

The deliverable is the Application Source Code Review report, which documents the application's existing security posture, identifies specific weaknesses and vulnerabilities, allowing development teams to correct application defects that otherwise weaken the robustness of the application from a security perspective.

Scope and Project Phases

Pre-Review Discovery

A representative from the target application's development team is asked to confirm the scope of the engagement. Trustwave requires a full list of the applications to be tested, design documentation, and third-party in-use, applications, and libraries used during the design, coding, and testing of the target application. This information allows the Trustwave consultants to become familiar with the existing application environment prior to the commencement of the engagement.

Documentation Review

Trustwave conducts a detailed review of the existing documentation for each application listed in this proposal, including design documents, concept of operations, and source code listings. On an as-needed basis, Trustwave requests clarification on components of the site, functionality, program flow, and design issues.

Architecture and Product Familiarization

Trustwave reviews the overall architecture of the application to become familiar with the security issues resulting from any third-party tools, applications, libraries, or services being used. This includes interface specifications for any pre-existing libraries or utilities, as well as security vulnerabilities or known issues with commercial tools and applications.

Static and Manual Source Code Analysis

The Trustwave team performs a detailed, manual analysis of the application source code. Many of the vulnerabilities discovered in a source code review are similar to vulnerabilities discovered during an Application Penetration Test. Unlike a penetration test, a code review allows for a greater breadth of coverage and an increased confidence level in the results of the assessment. This is principally a result of having a fuller understanding of the design, software architecture and its internals, allowing identified vulnerabilities to have their exploitability fully assessed from a risk perspective. Some vulnerabilities or design flaws are also far easier to discover in a code review, such as "hidden" functionality in an application, or deficiencies in auditing controls.

The following table lists some of the different vulnerability classes Trustwave covers during an Application Source Code Review. This list is not intended to be exhaustive and the actual review performed depends on the specifics of the application being assessed. Some of the vulnerabilities may not be possible in some types of languages or frameworks (e.g., managed memory environments) but Trustwave is experienced in providing security services for any architecture. In all projects, Trustwave customizes the test plan to fit the technology used by the application.

Copyright © 2018 Trustwave Holdings, Inc. All rights reserved.

Vulnerability Class	Items Tested
Access Control	• Inconsistent use of centralized authentication and authorization controls • Unlimited Login Attempts • Password Complexity Policy • Single-sign on integration • Inadequate Auditing Controls
Session Management	• Weak Session Identifier Generation • Session Replay • Session Fixation • Insufficient Session Expiration
Data Validation	• Improper Input Validation • Dynamic SQL Commands • Improper Output Encoding • Format Strings
Application Resource Handling	• Path Traversal • Predictable Object Identifiers • XML Entity Expansion • Local & Remote File Inclusion • Shell command execution
Cryptography	• Weak Algorithms • Poor Key Management • Insecure Data Storage
Memory Management	• Buffer Overflows (stack and heap-based) • Memory Leaks (leading to Denial of Service) • Null Allocations • Double De-allocation
Logical Attacks	• Abuse of Functionality • Workflow Bypass
“Hidden” Functionality	• Backdoors • Debugging Interfaces • Undocumented Inputs
Code Quality	• Verbose Error Messages • Unused / Dead Code • Improper Exception / Error Handling • Inconsistent Logging

The review team documents any potential security flaws. As and when questions arise regarding the operation or functionality of the code, Trustwave compiles questions and periodically contacts the development team to resolve any outstanding enquiries.

Deliverables

Following the conclusion of the engagement, findings are made available. The deliverables are both strategic and tactical in nature, presented in a format that is highly accessible to both management and operational staff. Each

Copyright © 2018 Trustwave Holdings, Inc. All rights reserved.

finding has an associated risk score, as well as containing detailed technical information pertaining to the nature of the finding. Each finding is also presented with clear guidance on how to remediate the issue.

Copyright © 2018 Trustwave Holdings, Inc. All rights reserved.