Secure Software Development Life Cycle

WatershedHealth

POLICY & PROCEDURE **SSDLC**		**POLICY #24**
SUPERSEDES POLICY:	**EFFECTIVE:**	**LAST REVIEWED:**
Security Compliance Program Security Policy 25 v.1	5/18/2022	5/10/2024

Purpose

The purpose of this Secure Software Development Life Cycle Policy (“SSDLC”) is to establish the information security practices for application development and configuration management for applications developed by Watershed Health (“Watershed”). The policy is intended to give direction on sensitive data security practices that are designed to ensure confidentiality, integrity, and availability of Watershed data.

Should Watershed ever elect to outsource software development, formal contracts will be in place to address the following:

1. Licensing agreements;
2. Code ownership
3. Intellectual property rights;
4. Certification of the quality and accuracy of the work;
5. Rights of access for the audit of the quality and accuracy of work;
6. Escrow arrangements;
7. Quality and security functionality requirements for the developed code; and
8. Security testing and evaluation prior to installation.

Applicability

This policy applies to all Watershed Workforce Members, computer systems managed by Watershed, and computer systems hosted by third-parties for which Watershed is responsible for the development of software and applications. The Product department is responsible for ensuring the implementation and maintenance of this policy.

Definitions

Information Security is defined as the protection of sensitive data environment information and its critical elements, including software applications, databases, cloud environments, data transmission protocols that store, use or process that information. Watershed uses a layered security model consisting of technical application security controls, secure software development process and quality assurance reviews, and third-party penetration tests to ensure data confidentiality, integrity, and availability. These strategic controls are intended to give direction on accepted security practices for the sensitive data environment.

Sensitive data means all individually identifiable information including financial information, account information, health information, and ePHI, and any information classified as Confidential, Protected, or Restricted, per Watershed Data Classification Policy. Sensitive data does not include de-identified data which has been de-identified in accordance with the guidance from the Department of Health and Human Services and NIST.

Policy

SSDLC Procedures shall be designed to comply with HIPAA and specific National Institute of Standards and Technology (NIST) recommendations, as defined in NIST special publication 800-53 and OWASP (Open Web Application Security Project) Guidelines, including OWASP ASVS (Application Security Verification Standards) guidelines.

Watershed will:

a. Ensure the confidentiality, integrity, and availability of all electronic protected health information ("ePHI") that Watershed creates, receives, maintains, or transmits through its applications and web interfaces.

b. Protect against any reasonably anticipated threats or hazards to the security or integrity of such information through security measures and patch updates to application configurations.

c. Protect against unauthorized access for any reasonably anticipated uses or disclosures of such information that are not permitted or required.

d. Ensure compliance with software development processes and procedures.

e. Require that all Business Associates agree to the same provisions of the HIPAA Security Rule when performing services in the fulfillment of the SSDLC Procedures.

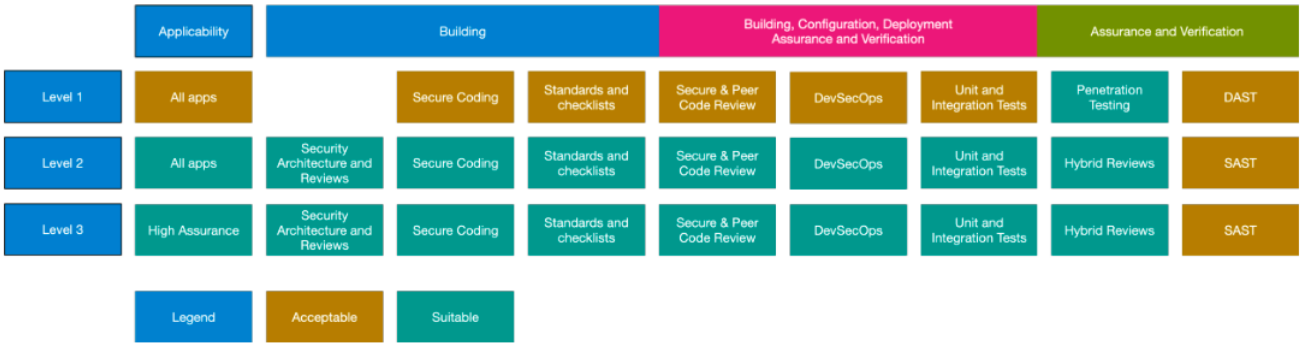
SSDLC Program

Information security of the sensitive data environment is a business issue. The objective is to identify, assess, and take steps to avoid or mitigate risk to sensitive data environments during application development. The OWASP Application Security Verification Standard (ASVS) defines three security verification levels, with each level increasing in depth.

- Level 1 is meant for all software and is typically appropriate for applications where low confidence in the correct use of security controls is required. Level 1 controls can be verified either automatically by tools or simply manually without access to source code.
- Level 2 is for applications that contain sensitive data, which requires protection. Level 2 ensures that security controls are in place, effective, and used within the application. Level 2 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 is for the most critical applications - applications that perform high value transactions, contain sensitive medical data, or any application that requires the highest level of trust. Level 3 requires more in-depth analysis, architecture, coding, and testing than all the other levels. A secure application is modularized in a meaningful way (to facilitate resiliency, scalability, and most of all, layers of security), and each module (separated by network connection and/or physical instance) takes care of its own security responsibilities (defense in depth), that need to be properly documented.

Each ASVS level contains a list of security requirements (see OWASP Application Security Verification Standard 4.0.3). Each of these requirements can also be mapped to security-specific features and capabilities that must be built into software by developers.

Table 1: Levels and controls for applications built



	Applicability	Building			Building, Configuration, Deployment Assurance and Verification			Assurance and Verification	
Level 1	All apps		Secure Coding	Standards and checklists	Secure & Peer Code Review	DevSecOps	Unit and Integration Tests	Penetration Testing	DAST
Level 2	All apps	Security Architecture and Reviews	Secure Coding	Standards and checklists	Secure & Peer Code Review	DevSecOps	Unit and Integration Tests	Hybrid Reviews	SAST
Level 3	High Assurance	Security Architecture and Reviews	Secure Coding	Standards and checklists	Secure & Peer Code Review	DevSecOps	Unit and Integration Tests	Hybrid Reviews	SAST
	Legend	Acceptable	Suitable						

Each sensitive data environment (e.g., the application) will have a designated Application Owner who has responsibility for the development, configuration, evaluation, and review of the processes and procedures throughout the life cycle. The Application Owner will ensure compliance with the roles and responsibilities, procedures, and associate quality assurance reviews and implement effective security controls within all phases of the software development life cycle. The **Secure Software Development Life Cycle Procedures** describe each of the following requirements for:

a. Allocation of resources to document security requirements and ensure alignment with Watershed business priorities;

b. Software development process to require developers to remove non-production application accounts, user IDs, and passwords before applications become active or are released to customers;

c. Review of custom code prior to release to production in order to identify any potential coding vulnerability;

d. Implementation of effective change control procedures;

e. Separation of development/test and production environments;

f. Separation of roles and responsibilities;

g. Implementation of secure coding guidelines including but not limited to OWASP top ten issues;

h. Development of security testing and evaluation criteria and procedures;

i. Performing integration and regression testing for components and services and unit, integration, and system testing for systems;

j. Performing threat and vulnerability analysis; and

k. Performing internal and independent penetration testing/analysis.

Watershed will retain documentation for each application developed for 6 years from the date of creation or the date when updates or changes were last in effect, whichever is later.

Procedures

a. **Roles and Responsibilities.**

   i. Security Officer or designee is responsible for:

      (1) Approving and issuing policies, procedures, and guidance for implementing the secure software development life cycle program.

      (2) Coordinating with the various Watershed stakeholders (i.e., Product, Security, Privacy, Compliance) throughout the development life cycle to address any environmental or operational changes or regulatory requirements.

      (3) Directing, monitoring, and enforcing implementation and maintenance of, and compliance with *Security Policy #25, Configuration & Change Management*; and

(4) Overseeing periodic testing and evaluation of Technology components to determine effectiveness and compliance with *Security Policy #25, Configuration & Change Management* as it relates to in-house developed applications utilized by Watershed and its Customers.

(i) This should include procedures for evaluating, assessing, or testing the security of externally developed applications utilized by Watershed or its Customers within the context of Watershed's technology environment.

ii. Allocation of Resources.

(1) Determine information security requirements for the information system or information system service in mission/business process planning;

(2) Determine, document, and allocate the resources required to protect the information system or information system service as part of its capital planning and investment control process.

b. **Software Development Process**. Watershed shall require developers to implement the following software development processes:

i. Establish password requirements in accordance with the *Security Policy #17, Person or Entity Authentication; and*

ii. Remove non-production application accounts, user IDs, and passwords before applications become active or are released to customers; and

iii. Review custom code prior to release to production or customers to identify any potential coding vulnerabilities.

c. **Change Control Procedures**. Watershed shall require developers to follow change control processes and procedures for all changes to system components in accordance with *Security Policy #25, Configuration & Change Management*. The process must ensure:

i. Separate development/test and production environments;

ii. Production data is not used for development but may be used for customer acceptance testing and quality assurance.

iii. Documentation of approval of changes to system components by Product Owner, CEO, or COO prior to production; and

iv. Monthly review of changes to and approval of system components by Cybersecurity Oversight Committee.

d. **Secure Coding Guidelines**. Watershed shall require developers to develop applications based on secure coding guidelines to prevent common coding vulnerabilities in software development processes, to include the following:

i. Injection flaws, particularly SQL injection (also consider OS Command Injection, LDAP and XPath injection flaws, as well as other injection flaws);

ii. Buffer overflow;

iii. Insecure cryptographic storage;

iv. Insecure communications;

v. Improper error handling;

vi. Vulnerabilities identified in the vulnerability identification process; and

vii. For web applications and web application interfaces:

(1) Cross-site scripting (XSS);

(2) Improper Access Control (such as direct object references, failure to restrict URL access, and directory traversal); and

(3) Cross-site request forgery (CSRF).

e. **Functional Properties of Security Controls**. Watershed shall require a description of the functional properties of the security controls to be employed in accordance with *Security Policy #25, Configuration & Change Management*.

f. **Design/Implementation Information for Security Controls**. Watershed shall require the developer to provide design and implementation information for the security controls to be employed that includes:

i. High-level design;

ii. Security-relevant external system interfaces; and

iii. Standardized inactivity time-out requirements.

g. **Services in Use**. Watershed shall require the developer to identify early in the system development life cycle, the functions, ports, protocols, and services intended for organizational use.

h. **Information System Documentation**.

i. Maintain documentation of the development/test and production environments including:

(1) Secure configuration, installation, and operation of the system, components, or services;

(2) Effective use and maintenance of security functions/mechanisms; and

(3) Known vulnerabilities and remediation.

i. **Security Engineering Principles**. Watershed shall apply information system security engineering principles in the specification, design, development, implementation, and modification of the information system.

j. **Develop Configuration Management**. Watershed shall require the developer to:

   i. Perform configuration management during system, component, or service (development, implementation, and operation) in accordance with *Security Policy #25, Configuration & Change Management*;

   ii. Document, manage, and control the integrity of changes to configuration items under configuration management;

   iii. Implement approved changes to the information systems;

   iv. Document approved changes to the system, component, or service and the potential impacts of such changes, and;

   v. Track security flaws and flaw resolution within the system, component, or service.

k. **Integration/Regression Testing.** Perform integration/regression testing as part of the software development life cycle.

l. **Threat and Vulnerability Analysis**. Watershed shall perform threat and vulnerability analyses and subsequent testing/evaluation of the as-built system, component, or service.

m. **Penetration Testing/Analysis**. Watershed shall require penetration tests to be performed.

n. **Vulnerability Management.**

   i. Vulnerabilities identified will be prioritized for remediation using the CVSS Score as follows:

Rating	CVSS Score
Low	0.1 - 3.9
Medium	4.0 - 6.9
High	7.0 - 8.9
Critical	9.0 - 10.0

Training

The Security Officer or designee is responsible for ensuring that all Workforce members involved in software development receive specific training as it relates to this and any associated procedures.

Enforcement

Violations of this policy may include suspension or loss of the violator's use privileges, with respect to Watershed's information systems, termination of employment or volunteer, intern, contractor status with Watershed. Additional civil, criminal, and equitable remedies may apply.

Documentation

The Security Officer or designee is responsible for ensuring this version of the policy, together with any forms and other documentation created or obtained in accordance with the policy, will be retained by Watershed for a period of at least 6 years plus the current year from the date of creation or the date when last in effect, whichever is later.

References

Regulatory Authority:

Internal:

- *Security Policy #17, Person or Entity Authentication*
- *Security Policy #25, Configuration & Change Management*

External:

- NIST Special Publications: http://csrc.nist.gov/publications/
    - SP800-53 (Rev5) Security and Privacy Controls for Federal Information Systems and Organizations
    - SP800-66 An Introductory Resource Guide for Implementing the Health Insurance Portability and Accountability Act (HIPAA) Security Rule
- OWASP ASVS (Application Security Verification Standard) 4.0.3

Document Control

APPROVED BY:		
Lisa Stanley	5/28/2024	DocuSigned by: Lisa Stanley 9418DCC7CF3D47D...
(Printed Name)	**(Date)**	**(Signature)**

REVISION HISTORY			
Date	**Author**	**Version**	**Comments**
5/16/2022	Scott Snodgrass	1.0	Implemented
5/20/2022	Lisa Stanley	1.0	Reviewed
5/17/2023	Nicole Montagnet	2.0	Reviewed and updated by Privacy & Security Officer
5/10/2024	Nicole Montagnet	2.0	Reviewed