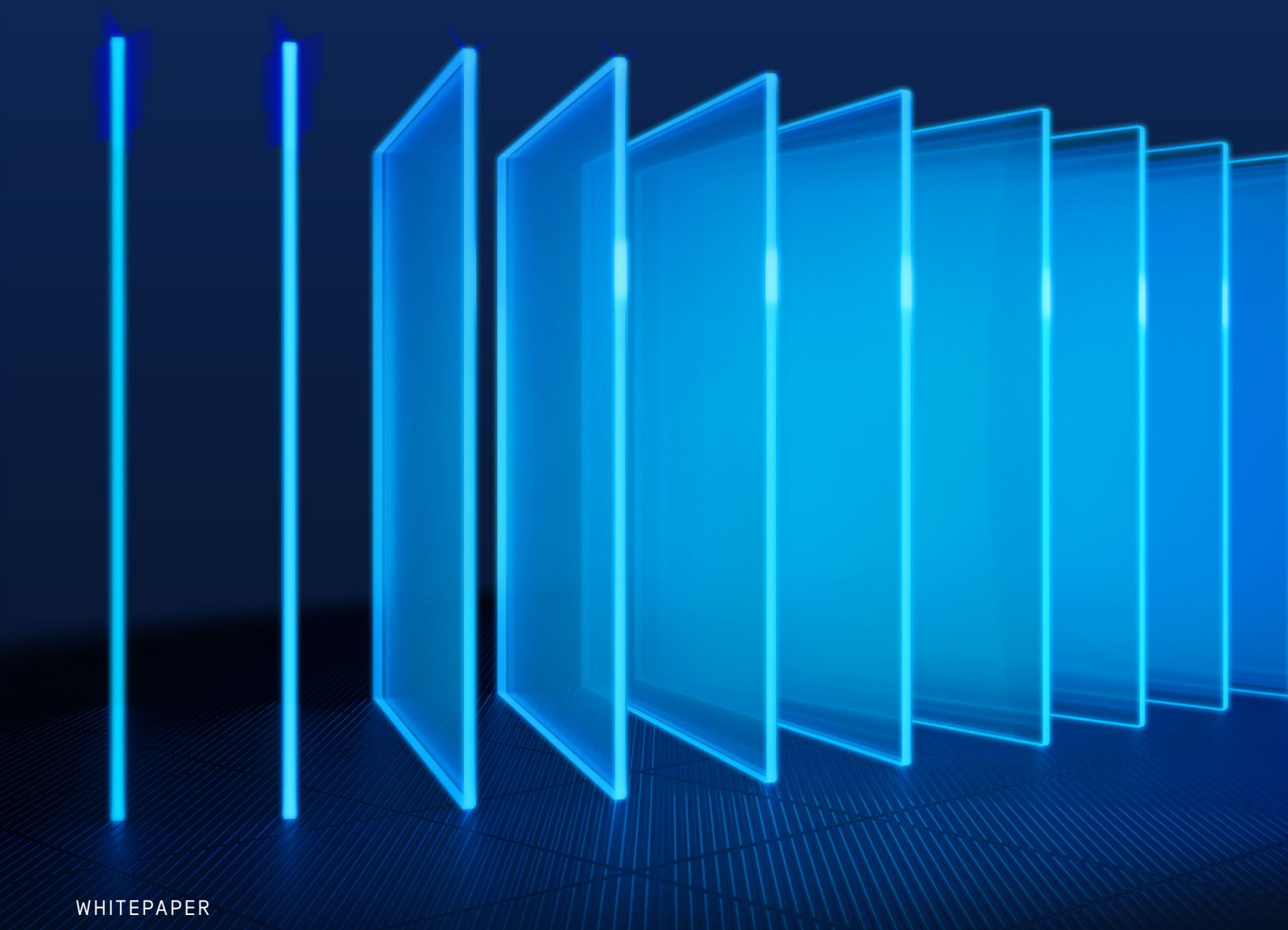
OPSWAT.



WHITEPAPER

Secure SDLC at OPSWAT

Executive Summary

This whitepaper defines OPSWAT's Secure Software Development Life Cycle framework, program, and process, outlining security requirements, compliance expectations, and governance. It serves as an internal policy for product teams in OPSWAT, a compliance expectation for vendors, and an informational guide for customers interested in our secure development practices.

Table of Contents

OPSWAT.

01. The Secure Software Development Lifecycle

What is Secure SDLC?

SDLC (Software Development Life Cycle) is a process consisting of a series of planned activities to develop software products. The Secure SDLC incorporates security into every phase of the Software Development Life Cycle—including requirement gathering, design, development, testing, and operation/maintenance.

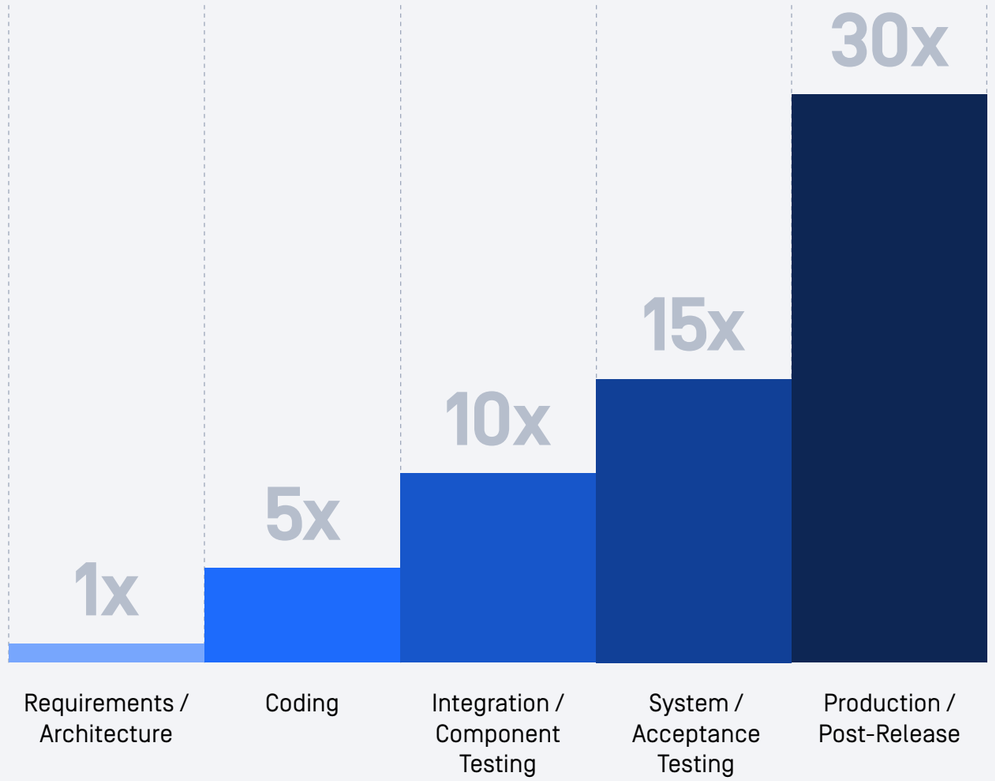
Why Secure SDLC?

Malicious actors target systems for profit or disruption, leading to costs, business risks, and reputational damage for organizations. According to a recent survey, the cost of fixing a security bug is 30 times higher when discovered in production versus during the analysis and requirements stage.

Implementing Secure SDLC provides the following benefits:

- Reduces business risk by detecting security flaws early in the development process.
- Reduces costs by addressing vulnerabilities early in the lifecycle.
- Establishes continuous security awareness among all stakeholders.

The Relative Cost of Fixing a Flaw at Different Stages of the SDLC



02. OPSWAT's Secure SDLC Framework

OPSWAT's Secure SDLC Framework defines structured methodologies and security principles that guide secure software development.

OPSWAT follows the Agile Software Development Lifecycle. To be fully compliant with our customers' requirements we adopted the Secure SDLC Framework to the regulatory requirements and international standards. This approach reinforces our commitment to continuous improvement and resilience in an evolving cybersecurity landscape.

NIST Secure Software Development Framework

OPSWAT's Secure SDLC Framework is built upon the NIST SP 800-218 SSDF (Secure Software Development Framework), ensuring that security is structured, measurable, and consistently applied across all stages of the software development process.

By integrating SSDF best practices, OPSWAT maintains a proactive security posture, embedding security into every phase of software development—from planning and design to implementation, verification, and continuous monitoring.

The attestation of compliance of individual products is provided to our U.S. Federal Government customers on-demand. See the contact details below.

ISO 27001 Information Security Management

Maintaining robust information security is critical to both operational integrity and regulatory compliance. OPSWAT's Secure SDLC Framework incorporates ISO 27001 ISMS (Information Security Management System) principles, ensuring that security controls, risk management strategies, and compliance measures are seamlessly integrated into the operation of our cloud products.

As both a provider and consumer of our security solutions, OPSWAT applies internally enforced company security policies, ensuring that our certified products adhere to enterprise-grade security expectations before deployment.

EU Cyber Resilience Act & the NIS2 Directive

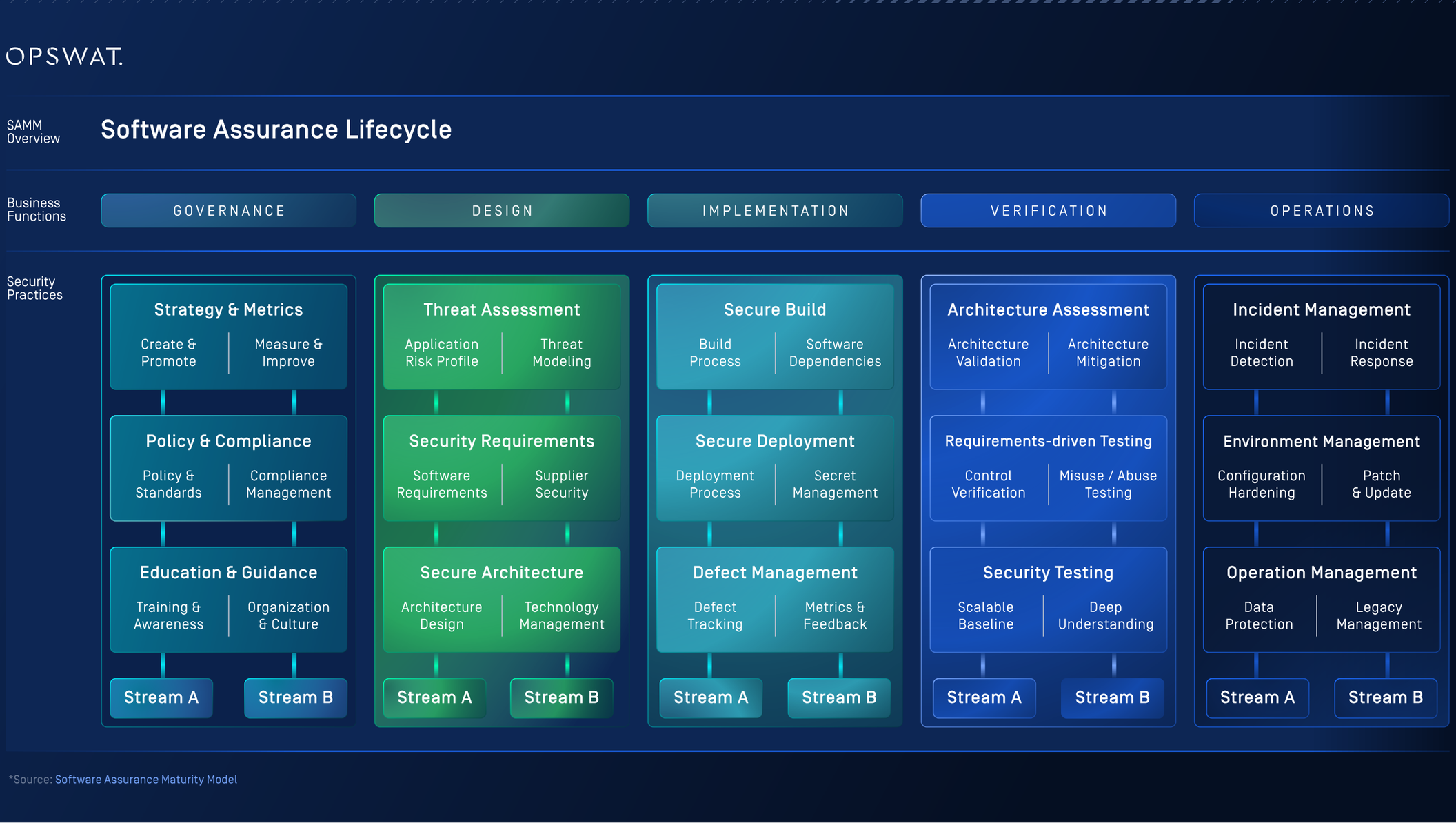
As cybersecurity regulations continue to evolve, OPSWAT remains committed to aligning its Secure SDLC Framework with global regulatory requirements, beginning with the EU Cyber Resilience Act and the NIS2 Directive. By proactively adapting to emerging standards, OPSWAT ensures that its Secure SDLC remains comprehensive, compliant, and resilient in an increasingly complex regulatory landscape.

ISO 9001 Quality Management

To ensure the highest standards of software quality, OPSWAT's Secure SDLC Framework is integrated into the ISO 9001 QMS (Quality Management System). The QMS establishes audited quality controls for governance, change management, and cross-functional processes, supporting the definition, design, development, production, and maintenance of the product and support offerings, extending beyond R&D to areas such as sales, customer support, information technology, and human resources.

This approach reinforces our commitment to a structured, risk-based approach to quality management, ensuring application security remains an integral consideration across all business functions.

See more details on Compliance & Certifications.



*Source: Software Assurance Maturity Model

OWASP Software Assurance Maturity Model

The OWASP SAMM (Software Assurance Maturity Model) is a comprehensive framework designed to help organizations assess, formulate, and implement effective software security strategies within their existing SDLC.

As an open-source framework, SAMM benefits from global contributions, ensuring a collaborative, continuously evolving approach to application security. Its structured methodology enables organizations with an effective and measurable way to analyze and improve their development lifecycle. SAMM supports the complete development lifecycle. SAMM is technology and process agnostic. SAMM is evolutive and risk driven. By leveraging SAMM, teams gain actionable insights into security gaps and can systematically enhance their security posture throughout the development lifecycle.

OPSWAT.

OWASP Application Security Verification Standard

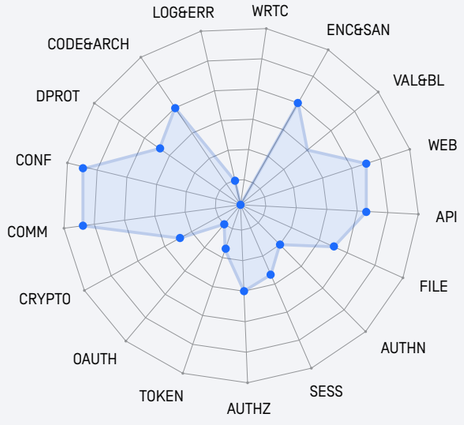
The OWASP ASVS (Application Security Verification Standard) is a globally recognized framework designed to establish a structured, measurable, and actionable approach to web application security. It provides developers and security teams with a comprehensive set of security requirements and verification guidelines to ensure that applications meet industry best practices.

As an open-source framework, ASVS benefits from broad contributions from the global security community, ensuring that it remains up to date with emerging threats and evolving security standards.

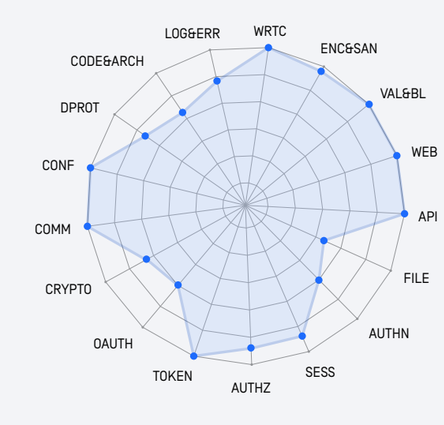
ASVS serves as a benchmark for application security maturity, enabling organizations to quantify security posture and systematically improve their secure development practices. With detailed security checklists covering critical areas such as authentication, authorization, session management, and access control, ASVS offers product teams clear, actionable guidance to integrate security seamlessly throughout the software development lifecycle. By adopting ASVS, organizations can enhance security assurance, streamline compliance efforts, and proactively mitigate vulnerabilities in modern web applications.

ASVS acts as a metric that provides application developers and owners with a standardized means to assess the level of security and trust in their applications. It also serves as guidance for security control developers, outlining the necessary security measures required to meet application security requirements. ASVS is a reliable basis for defining security verification requirements in contracts.

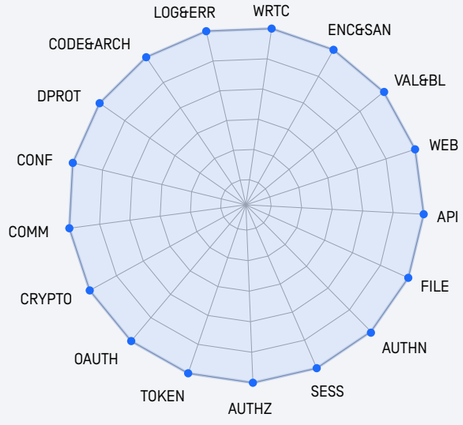
Level 1



Level 2



Level 3



Note	
ENC&SAN	Encoding and Sanitization
VAL&BL	Validation and Business Logic
WEB	Web Frontend Security
API	API and Web Service
FILE	File Handling
AUTHN	Authentication
SESS	Session Management
AUTHZ	Authorization
TOKEN	Self-Contained Tokens
OAUTH	OAuth and OIDC
CRYPTO	Cryptography
COMM	Secure Communication
CONF	Configuration
DPROT	Data Protection
CODE&ARCH	Secure Coding and Architecture
LOG&ERR	Security Logging and Error Handling
WRTC	WebRTC

03. Application Security Governance and Training

OPSWAT's Secure SDLC Program translates the Secure SDLC Framework into structured governance, ensuring that security requirements are documented, maintained, measured, and continuously improved, while also ensuring that all involved parties receive adequate training. It establishes roles, responsibilities, and security measures for development, testing, and production environments, as well as pipeline security, defining the Secure Development Environment and mandating the application of security policies within the Secure SDLC Process.

1. Roles and Responsibilities

High-Level Management – Chief Product Officer

The CPO (Chief Product Officer) is responsible for strategic oversight and enforcement of the Secure SDLC Program across all product teams as well as other R&D programs such as the QA Program and the UX Program, ensuring a cohesive approach to secure, high-quality, and user-centric software development.

As the primary risk owner for all products and R&D processes, the CPO mandates R&D Operations to own the Secure SDLC Program and ensures that product leaders are enforcing the application of the Secure SDLC Program and implementing the Secure SDLC Process effectively in the product teams. In this role, the CPO approves the modification of the Secure SDLC Program, and the deviations from the Secure SDLC Process.

The CPO also monitors the Secure SDLC Program outcomes, tracking security maturity, vulnerabilities, compliance, and development activities to maintain a strong security posture of the products.

Additionally, the CPO is responsible for R&D security budget allocation and approval, ensuring that adequate resources are dedicated to the Secure SDLC Program.

R&D Operations

The R&D Operations team is composed of software engineering leaders and application security engineers, ensuring compliance with regulatory and security requirements. The head of R&D Operations is the risk owner of both the Secure SDLC Framework and the centralized services of the Secure Development Environment, overseeing their continuous improvement and integration into OPSWAT's development processes.

As the owner of the Secure SDLC Program, R&D Operations is responsible for maintaining and evolving the program in coordination with the company security policies and the other R&D Programs. This includes aligning with product leaders on strategic roadmaps, defining and tracking Security KPIs to enhance maturity levels annually, and adjusting ASVS requirements as necessary.

Collaboration is central to this role, as R&D Operations organizes the Application Security Virtual Team, supports product teams in executing the Secure SDLC Program, verifies and reports on all product security postures, ensures ongoing security training, and provides expert guidance on application security best practices.

Additionally, R&D Operations manages the centralized services of the Secure Development Environment, ensuring compliance with company security policies, acting as custodians of the source code, and overseeing the configuration of CI/CD (Continuous Integration/Continuous Deployment) tools. This includes managing evidence collection within the CI/CD pipeline and enforcing strict access controls.

Product Teams

The product team is composed of the product leader, software engineers, developers, QA engineers, SREs (site reliability engineers), and other team members in various roles, depending on the specific needs of the product.

The product leader is the risk owner for their respective product, overseeing all team members and ensuring that the development process adheres to the Secure SDLC Process. The team is responsible for executing and implementing the OPSWAT Secure SDLC Program, ensuring security is integrated throughout the development process.

The team may customize processes, tools, and the CI/CD pipeline, defining release criteria and integrity measures while documenting any deviations from the Secure SDLC Process. A security champion is designated within the team, responsible for attending security-related meetings of the Application Security Virtual Team and ensuring effective communication within the team regarding security matters.

Additionally, the team is responsible for reporting evidence of the product's security posture, maintaining transparency, and ensuring continuous compliance with security standards.

Application Security Virtual Team

The Application Security Virtual Team is a cross-product team composed of application security engineers from R&D Operations and designated engineers serving as security champions from each product team, all focused on ensuring the security of OPSWAT's products.

During regular meetings, security champions receive updates on topics such as security KPI changes and the recommended use of security-related CI/CD tools in the pipeline. These meetings also provide a forum for the parties to share their experiences, discuss security-related issues, and initiate the Secure Review process. Additionally, they actively participate in RCA (root cause analysis) to improve security posture and prevent recurring vulnerabilities.

2. Security Program Strategy

Strategic Priorities

OPSWAT's strategic plan for application security is aligned with its business priorities and risk appetite, considering the maturity level of each product and its exposure to security threats. The primary focus is on safeguarding high-risk products, particularly those with a large customer base, public-facing deployments, or integration into critical infrastructure.

Security Budget

A dedicated security budget under R&D Operations is allocated for key security initiatives and tools, including third-party audits, independent penetration testing, and automated security testing within the CI/CD pipeline.

Automation and Independent Verification

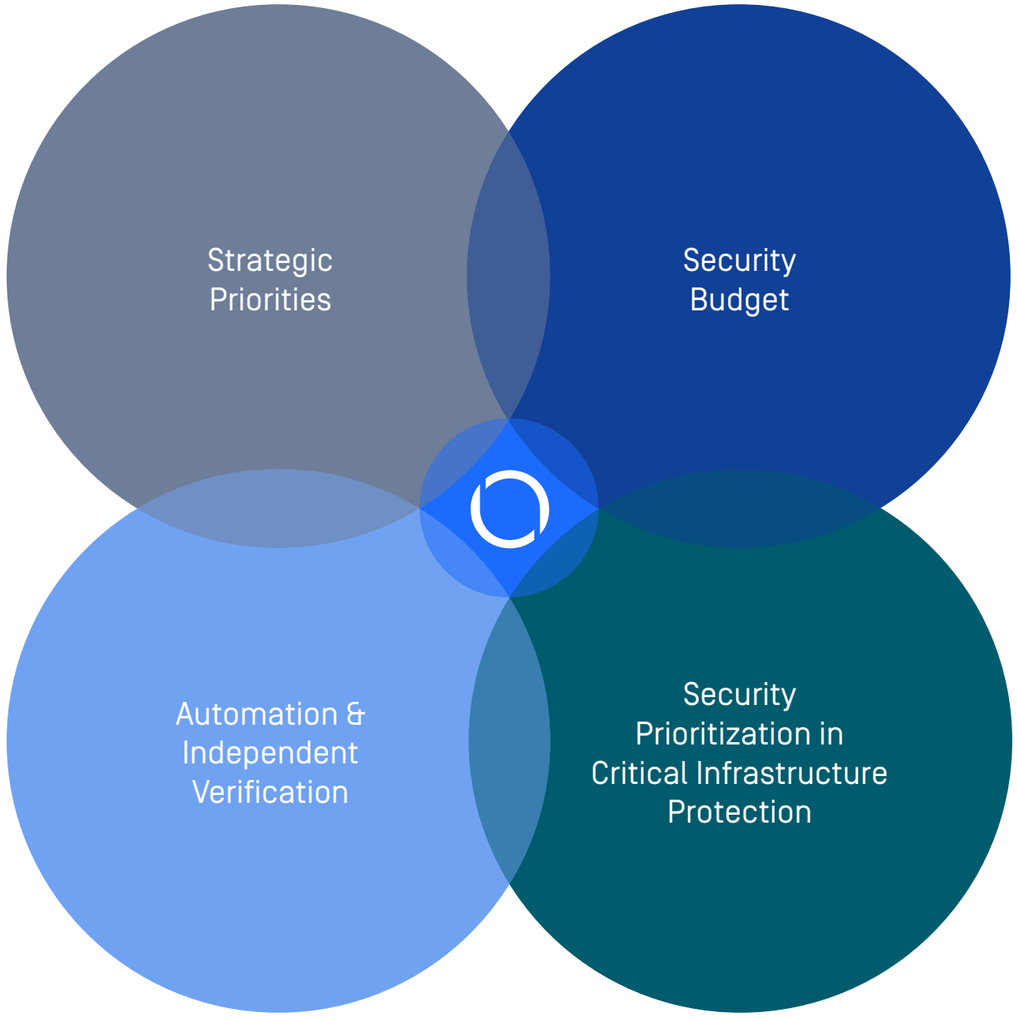
To minimize product security risks, OPSWAT prioritizes preventive security measures based on risk assessments. This includes the integration of automated security scanning within the CI/CD pipeline orchestration, enabling early detection and remediation of vulnerabilities throughout the development lifecycle.

Additionally, internal assessments, third-party audits, and independent penetration testing reinforce security by eliminating single-point dependencies and ensuring a structured, multi-layered verification process. This approach strengthens risk identification and mitigation efforts, ensuring that vulnerabilities are comprehensively addressed and validated by independent security professionals.

Security Prioritization in Critical Infrastructure Protection

In the context of CIP (critical infrastructure protection), security remains the highest priority, particularly in the rare cases when it conflicts with regulatory requirements or quality attributes. Decision-making follows these guiding principles:

- Security takes precedence over regulatory conflicts related to privacy, environmental, or sustainability regulations.
- Security and reliability outweigh other quality attributes such as usability, maintainability, and compatibility (as per ISO/IEC 25010).
- Integrity and availability take priority over confidentiality in cases where system reliability is more critical than restricting access (as per ISO/IEC 27001).



3. Security Training and Awareness

As part of the Secure SDLC Program, in addition to the company's general security awareness training, role-specific security training is mandated for all staff involved in secure development. All training courses are tracked in the company's training tools. Training and awareness programs are reviewed periodically to incorporate new security trends and ensure ongoing compliance with security standards.

Awareness Initiatives

- Security testing of infrastructure and personnel in alignment with company security initiatives.
- Internal vulnerability scanning of products and infrastructure.
- Daily internal and external network scans.
- Social engineering campaigns.

Role-Specific Trainings

- Training campaigns for product teams, covering OWASP Top 10, API security testing, and cloud security training.
- Training campaigns for product teams on the policies outlined below.
- Developers participate in continuous secure coding training through a dedicated learning platform.

Onboarding

- New employee onboarding includes all relevant security training based on their role.
- Security champions undergo specific onboarding training when they join the Application Security Virtual Team.

4. Measurement and Continuous Improvement

OPSWAT is committed to continuously enhancing its Secure SDLC Program through structured performance measurement, maturity assessments, and regular updates to ensure ongoing security effectiveness.

To maintain a strong security posture, OPSWAT employs a systematic approach to tracking and improving security performance. This includes quarterly product security maturity assessments, internal security reviews to verify adherence to best practices, and the definition of annual KPIs (Key Performance Indicators), which are measured quarterly.

To effectively measure application security posture, OPSWAT evaluates teams using structured metrics. The product security maturity is assessed per team based on the SAMM framework, providing a quantifiable measure of security progress. In addition, products undergo an ASVS compliance assessment to ensure adherence to security verification requirements. Compliance with the Secure SDLC Process is closely monitored, assessed, and the achievement of KPI goals is evidence-based, ensuring that the security posture and security improvements are both measurable and actionable. All product teams are required to meet security maturity targets as part of their annual performance evaluation.

As part of its continuous improvement efforts, OPSWAT introduces new product security initiatives periodically to increase maturity levels and strengthen application security. These initiatives include updating security policies to address emerging threats, integrating new security tools for enhanced detection and prevention, and expanding KPI objectives to drive ongoing progress.

To further reinforce security governance, OPSWAT conducts an annual review of the Secure SDLC framework, incorporating insights from root cause analyses of past security incidents, assessments of vulnerability trends, and refinements to existing processes and policies.

This structured approach to continuous improvement ensures that OPSWAT maintains a proactive and resilient product security posture, effectively adapting to evolving cybersecurity challenges while meeting both regulatory and operational security objectives.

5. The Secure SDLC Process

The Secure SDLC Process further operationalizes the Secure SDLC Program by defining the security controls that teams must follow, including specific activities such as automated security checks and verification mechanisms at each development phase. This process is aligned with other key R&D programs, such as the Quality Assurance Program and the User Experience Program, ensuring a cohesive approach to secure, high-quality, and customer-focused software development. The Secure SDLC Process is detailed in Sections 4-9.

The Secure SDLC Process is a high-level process, teams may implement it in an extended customized way with the condition of the security of the process must be kept on the same level as minimum. Deviation from the Secure SDLC Process must be documented and approved.

6. Policies Under the Secure SDLC Program

The Secure SDLC Program comprises various policies that must be formally approved and acknowledged by product teams to ensure compliance with its requirements. Adherence to these policies is mandatory internally, and each team is responsible for reviewing, signing, and implementing them as part of their development processes.

Below is a list of key policies along with their respective purposes. For policies with external significance, additional details are incorporated into this document.

Policy	Description
Application Security Verification Policy	This policy defines the verification of the products' security in detail, see more details in the Application Security Testing and Verification section.
Release Integrity Policy	This policy defines the code signing requirements, see more details in the Release Integrity section.
SBOM Management Policy	The SBOM (software bill of materials) management policy purpose is to ensure the up-to-date status of the used third-party components registry. This is the basis of other policies that deal with third-party legal and security risks.
Supply Chain Security Policy	This policy defines the conditions of usage of open-source or third-party components, and a process to introduce new open-source or third-party components, including vendor assessment, see more details in the Vendor Assessment section.
Product Vulnerability Management Policy	This policy defines the remediation timeframes for open-source, third-party, and internal vulnerabilities and establishes procedures for handling security patches across all products. It ensures that vulnerabilities are assessed, prioritized, and resolved within defined timelines.
End-of-Life Component Management Policy	EOL (End-of-Life) components pose a security risk and are therefore not permitted to be used in our products. This policy outlines the management of unexpected situations that arise when a component reaches end-of-life.
Product Privacy Compliance Policy	This policy defines the privacy compliance requirements for the products and the appropriate security controls to be applied.
Malware Samples Handling Policy	This policy defines the procedures for safely handling live malware samples to prevent malware incidents in our environments.
AI Usage Policy	The AI (Artificial Intelligence) Usage Policy restricts the use of AI in development to ensure the security of our customers. AI serves solely as an assistive tool, while individual developers remain fully responsible for the development process. AI tools may only be used in private mode, strictly preventing any exfiltration of source code or other security-related information.
Product Vulnerability Disclosure Policy	This policy defines roles and responsibilities for managing vulnerabilities, covering the entire lifecycle from detection and remediation—as outlined in the Product Vulnerability Management Policy—to coordinated disclosure, see more details in the Secure Operation and Maintenance section.

04. Secure Design and Risk Assessment

As part of the Secure SDLC Process, security requirements are tracked, documented and maintained throughout the development lifecycle. Third-party vendors are required to acknowledge and meet ASVS, ensuring consistency in security expectations and adherence to the Product Privacy Compliance Policy across all software components.

Security is incorporated into every phase of the development lifecycle. It is the responsibility of the security champions to keep in mind the expectations of the Secure SDLC Process and represent them within their teams.

The Secure Design requirement set includes ASVS-based functional and non-functional security requirements. Reference models are provided by R&D Operations to support design decisions, along with documented adjustments to ASVS requirements if needed (e.g., stronger encryption requirements).

Threat Modeling

Threat modeling is a structured process for identifying threats and vulnerabilities at the earliest stages of the development life cycle. It is an integral part of the Secure SDLC Process, conducted regularly—at least once a year or whenever new features or architectural changes are introduced. Product teams perform threat modeling by defining security objectives, identifying assets and dependencies, analyzing potential attack scenarios, and mitigating identified threats.

An enhanced approach incorporates data flow analysis and established threat modeling practices (e.g. the STRIDE model), ensuring a comprehensive evaluation across products. When necessary, Security Reviews are initiated to validate compliance with security requirements and proactively address potential risks. Design decisions are carefully documented, and any remaining risks are continuously tracked throughout the product lifecycle.

Risk Assessment and Mitigation

Application security risks are assessed using multiple sources, including residual threats identified during threat modeling, widely recognized security vulnerabilities such as those in the OWASP Top 10 and SANS Top 25, and missing security controls based on ASVS guidelines. Additional risk factors include weaknesses in secret management throughout the build, deployment, and release processes, as well as vulnerabilities in open-source and third-party components.

Following the risk assessment, mitigation plans are developed to reduce the severity of identified risks, taking both impact and likelihood into account. These plans, along with the corresponding risks and mitigation steps, are thoroughly documented.

Residual risks are tracked throughout the product lifecycle, and subject to periodic review and must be formally acknowledged by the risk owners. They are also incorporated into internal release reports to maintain visibility and accountability.

When necessary, Security Reviews are initiated to ensure compliance with security requirements and to proactively address potential risks, reinforcing the overall security posture of the product.

Secure Design Best Practices

Secure Design Principles are collections of desirable product properties, behaviors, designs and implementation practices.

The product team must apply the security functionality related principles such as Least Privilege, Fail Securely, Establish Secure Defaults and Least Common Mechanism.

The product team must apply the secure software architecture related principles such as Defense in Depth, Principle of Open Design and Leverage Existing Components.

The product team should apply the user experience related principles such as Psychological Acceptability and Economy of Mechanism in the design in consistency with the User Experience Program.

Product teams must follow these and all other state-of-the-art principles necessary to prevent security flaws in the architecture and security or non-security features.

To support the product teams in the implementation of Secure Design Principles, the R&D Operations provide several guidelines based on the principles with security reference models for critical security features.

The product team is required to create a Security Test Plan consistent with the Quality Assurance Program, defining the security test cases for functional and non-functional security requirements including tests for misuse and abuse cases, the test data, including attack patterns (e.g. DOM based cross site scripting, Cross Site Scripting Injection) and the testing tools.

05.
Secure Implementation, Build and Deployment

As part of the Secure SDLC Process, the Secure Implementation, Build, and Deployment phases aim to prevent vulnerabilities and flaws, based on the Secure Design and Risk Assessment. The requirement set contains expectations on ASVS based functional and non-functional security requirements, secure development, and test methodology relying on the Secure Development Environment.

During the implementation, the Secure Coding Best Practices, Secure Code Review, and Early Detection of Security Flaws are to be applied. Teams must adhere to Supply Chain Security Policy (including vendor onboarding and open-source software topics), AI Usage Policy, and Malware Samples Handling Policy. During the build and deployment Secure Build and Deployment with centralized CI/CD pipeline usage and Separation of Duties are required.

Secure Coding Best Practices

Product teams must follow language-independent secure coding best practices during implementation. They are required to validate input data, sanitize data sent to other systems, eliminate compiler warnings, set secure error messages, apply output encoding where applicable, implement secure logging without exposing sensitive data, and follow proper error handling and exception management guidelines. Teams must also ensure that cryptography, if used, relies on approved algorithms and secure random number generation, and securely manage system resources by handling memory safely, preventing race conditions, and avoiding deadlocks through proper synchronization.

Product teams are also advised to follow language-specific secure coding guidelines, enforced by SAST tools, as exemplified below:

For Java, teams should ensure that keys used in comparison operations are immutable, use SecureRandom instead of Random, and avoid insecure deserialization by validating or restricting input classes.

In C++, it is recommended to detect and handle memory allocation errors, prevent buffer overflows through bounds checking and the use of smart pointers such as std::unique_ptr(), and avoid unsafe functions like strcpy() and sprintf().

For Python, developers should avoid using functions like eval() or exec() to mitigate code injection risks and prefer secure serialization formats such as the json module over pickle when processing untrusted data.

Secure Code Review

As part of the Security Reviews required by the Application Security Verification Policy, secure code review is important and executed depending on the development technology with various Secure Code Review Checklists are applied based on OWASP Cheatsheet series.

Early Detection of Security Flaws

As required by the Application Security Verification Policy, early detection of security flaws is a critical component of the development process. To minimize potential security issues, a “fail to build” approach is mandatory, ensuring that insecure code does not proceed through the pipeline. Additionally, a “fail to merge” approach is enforced, requiring teams to remediate any detected issues before changes can be integrated. Resolving the detected flaws is essential to meeting the release criteria.

Secure Build and Deployment

As part of the Secure SDLC process, the use of a centralized, orchestrated CI/CD pipeline is mandatory to enforce secure builds and avoid supply chain attacks. Audit, build, and deployment logs are generated, preserved, and reviewed as defined in the company security policies.

Every product team is responsible for following secure build and compiler configurations where applicable. They must use secure compiler options, disable debug code, harden runtimes for interpreted languages, pin dependency versions, ensure reproducible builds, and harden container images. The configurations used must be documented and periodically reviewed.

In alignment with the Separation of Duties principle, developers and other team members who have code or build access cannot have access to the production environment. In the case of cloud products, only the product’s site reliability engineers are allowed to deploy to the production environment.

Leveraging Existing Components

The product teams adhere to industry best practices for specific security functions (e.g., FIPS 140-3 compliant cryptography). In alignment with the Open Design principle, we use widely accepted open-source components for these security features.

To ensure third-party components remain up to date, we adhere to our End-of-Life Component Management Policy.

Internally developed components, whether for internal use or as subcomponents in other products, must follow the Secure SDLC Process and meet the same security requirements.

Our cloud products utilize common, internally developed components to implement specific security features.

OPSWAT.

06.
Application Security Testing and Verification

According to our Application Security Verification Policy we implement formal documentation and tracking for discovered issues and assign automated tools for continuous verification. As part of the Secure SDLC Process security checks are enforced and tracked at every stage of the SDLC to meet compliance requirements. The purpose of these is to find efficiently the possible security flaws. The arising security issues are investigated by the teams and addressed within the timeframe. The timeframes are part of the defined security KPIs.

Security Reviews

- **Architecture and Design Reviews:** Senior engineers and members of the Application Security Virtual Team assess security aspects in design changes, including encryption, authentication, authorization, auditing, system hardening, and system and network architecture.
- **Code Reviews:** On top of the regular code reviews by peer and senior engineers, members of the Application Security Virtual Team review the changes to prevent common flaws like injection, error handling, and insecure configurations.

Early Detection of Security Issues

- **Secret Scanning** to avoid secret exfiltration and ensure good design and secure implementation of secret handling.
- **SAST (Static Application Security Testing)** tools to detect vulnerabilities (e.g. SQL Injection, Buffer Overflows).
- **SCA (Software Composition Analysis)** is used to detect open-source vulnerabilities.
- **DAST (Dynamic Application Security Testing)** is used to find runtime (e.g. memory flaws) and environment issues.

The tools that are defined in the Early Detection of Security Issues section are mandatory to use in the CI/CD pipeline. All identified vulnerabilities must be fixed following the Product Vulnerability Management Policy.

Security Testing

Both automated and manual security testing methodologies are used together with the Quality Assurance Program executing the Security Test Plan.

- DAST tools are used to detect run time vulnerabilities, test default configurations, and test the system resilience after applying the hardening suggestions. The tests target both the software and the underlying infrastructure.
- To avoid regression in security requirements and features, we use automated testing tools to continuously verify the integrity of security features and controls.
- Manual testing is applied where automated tools fall short, such as in verifying controls for information leakage, identifying business logic flaws, and contextual vulnerabilities.
- The automated Malware Scanning of artifacts in the development lifecycle is also part of the steps that focus on the prevention of security issues.

Penetration Testing

Penetration testing is performed regularly and on demand both by internal penetration tester team and independent external vendors. Security champions triage vulnerabilities found to determine whether issues require code or configuration changes. For vulnerabilities that require code changes, product backlogs are created and resolved as quickly as possible.

The penetration test report for individual products is provided to our customers on demand. See the contact details below.

OPSWAT.

07. Secure Releasing

As part of the Secure SDLC Process, the release process enforces release criteria ensuring both adherence to Secure SDLC Process and the overall security of the product, based on the findings according to the Application Security Testing and Verification. Product versioning plays a crucial role in maintaining security improvements across releases, preventing security related regression, and preserving the achieved security posture, as a fundamental requirement for each release.

The release process includes the generation of internal release reports, which document residual risks and any outstanding security issues. These reports must be formally approved by the product leader. Additionally, external release notes communicate security-related changes and fixes as part of the product's official release.

For cloud products, deployment follows a "fail to deploy" automation approach, ensuring that only secure builds are released. Application Security Testing and Verification is integrated into the deployment pipeline, with an operational pull strategy rather than a push, reinforcing security validation before production deployment.

Per the SBOM Management Policy, each release includes an SBOM to maintain traceability of component provenance, supporting transparency and supply chain security. All necessary release files are securely archived to ensure long-term accessibility.

Release Integrity

According to the Release Integrity Policy, to uphold the integrity and security of product releases, a structured versioning system (e.g. Semantic Versioning) is applied, ensuring clear traceability of changes and a defined retention period for all released artifacts, including documentation. To further enhance security, software artifacts are digitally signed under the company's name, with published SHA fingerprints allowing users to verify authenticity and detect any tampering attempts.

Versioned documentation accompanies each release, providing detailed guidance on integrity verification methods, secure installation procedures, configuration best practices, and system hardening measures. These resources help users implement security controls effectively, reducing potential attack surfaces. Additionally, the EULA (End User License Agreement) is included to establish compliance obligations and maintain legal transparency.

The SBOM for individual products is provided to our customers on demand. See the contact details below.

08. Secure Operation and Maintenance

As part of the Secure SDLC Process in Operations and Maintenance, all products and services must comply with company security policies, including adherence to the Security Incident Response Plan and, where applicable, the BCP (Business Continuity Plan).

The operation of cloud production environments falls under the responsibility of the SRE team. In accordance with the Separation of Duties principle, the SRE team members who have access to production environments do not have access to development environments, including the source code and build pipeline.

The SRE team is continuously updating the infrastructure with security patches and upgrading it to align with LTS (Long-Term Support) versions provided by vendors or delivered by product teams, in accordance with the End-of-Life Component Management Policy.

We adhere to a Product Vulnerability Disclosure Policy that defines roles and responsibilities in managing security vulnerabilities.

SRE team triages security incidents affecting products with the involvement of security champions if needed.

Built around the Product Vulnerability Management Policy, this policy extends the R&D remediation process by incorporating:

- External vulnerability and incident reporting, ensuring prompt handling of reported issues.
- Internal incident reporting is triggered when necessary based on severity.
- RCA must be conducted after any major or recurring security incident to identify recurring issues and prevent future vulnerabilities.
- Secure SDLC updates, implemented when necessary to strengthen security measures.
- Once remediation is complete, a coordinated vulnerability disclosure, ensuring transparency.

To report a vulnerability found by external parties, see the contact details below.

09. Secure Development Environment

Development, testing, and production environments are securely separated to prevent unauthorized access. Each environment follows strict hardening baselines and endpoint security protocols. Development Environments must be compliant with the company security policies.

Endpoint Protection

As part of the endpoint protection all OPSWAT owned devices are monitored for vulnerabilities, installed software, installed patches, and compliance with the company security policies. In case of non-compliance, restrictive actions are taken to limit access to corporate resources.

Resources classified with high-risk category can be accessed only via controlled access routes (VPN). Devices outside the company network are forced to use secure channels to access R&D resources.

Pipeline Security

The CI/CD pipeline security adheres to strict security directives to mitigate evolving threats. The source of the threats could be outdated infrastructure elements (like operating systems, analysis tools, etc.), unauthorized access due to weak privilege controls and poorly isolated environments. Keeping the CI/CD infrastructure up-to-date, thoroughly vetted, and tightly controlled is a cornerstone of our secure SDLC.

Regionally, U.S.-based servers are used for all centralized services, including code storage, the CI/CD pipeline, analysis and testing tools, and secure artifact signing. The configuration of all centralized tools is under the control of R&D Operations.

We apply strong authentication mechanisms (Multi-Factor Authentication - MFA) and authorization controls (Role-Based Access Control - RBAC). Least privilege and regular access reviews are conducted.

Our pipelines incorporate several analysis and testing automation tools, including SAST, SCA, DAST, Secret Scanning, and Malware Scanning.

In our secure code signing solution, we use HSMs (Hardware Security Modules) to protect the key material against unauthorized access and to generate the signature. The signing solution is part of the CI/CD infrastructure, but network segmentation is in place. Only R&D Operations is authorized to access the HSMs for short periods. Every signing action is logged and can be reviewed during an audit trail.

The toolset used to build, compile, or test the software must have provenance information and come from a validated source. The tools used in the CI/CD pipeline are limited in number; only the necessary tools are installed. Only LTS software is allowed for compiling and building steps in the pipeline. In the operation of the centralized services, regular maintenance and key rotation periods are defined. Internally developed tools fall under the Secure SDLC Process.

Environment hardening for all centralized services is continuous, and these security requirements are reviewed periodically. Hardening guidelines are communicated to the product teams to ensure they are prepared and may tailor their development processes accordingly. In the event of a security incident, a RCA is conducted to take preventive action and update these requirements.

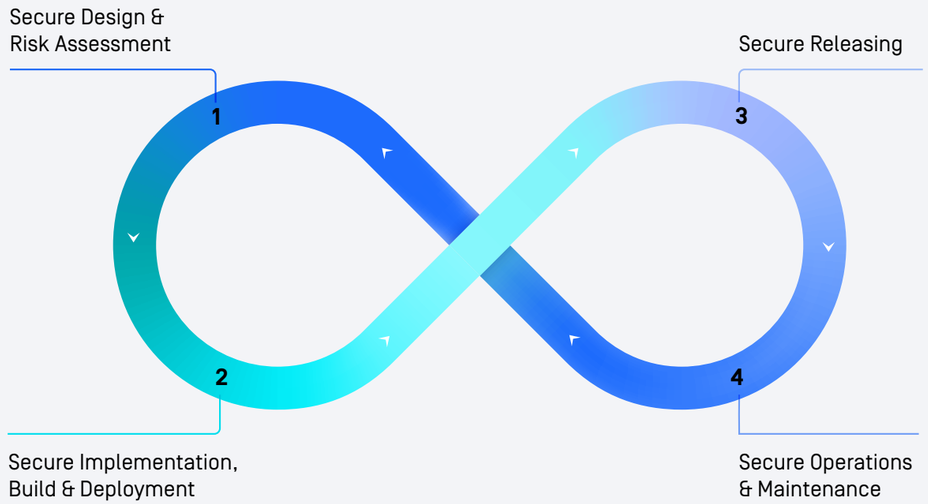
Code Protection

The protection of source code is a crucial part of the software development to guarantee the confidentiality and integrity of source code within the company.

Source code is stored following the principle of least privilege, allowing access only to authorized personnel and tools. The source code is under version control. The version control management system guarantees the traceability and accountability of code changes. Source code storages are encrypted with FIPS 140-3 compliant cryptography and protected with an appropriate key length.

Vendor Assessment

As part of our Vendor Onboarding Process, vendors are subject to a Sanctions check. As part of our contracts with vendors and suppliers, they are also obliged to maintain regulatory compliance throughout the term of the contract, including maintaining adequate export licenses under the EAR (Export Administration Regulations), when applicable. The vendor assessment process may include evaluation checklists, security and privacy reviews, and a review of third-party audits, and certifications. Critical vendors are reviewed and assessed at least annually. Any non-compliance with our expectations is tracked and a risk assessment is conducted in such cases.



OPSWAT.

10. Conclusion

Internal Application of the Secure SDLC

Compliance with this policy is mandatory for all internal teams. This document is subordinate to company policies, meaning that in the event of any contradictions, company policies take precedence and must be followed.

Escalation process for Secure SDLC violations: Any violations of this policy are handled internally, starting with R&D Operations and escalating up to the CPO as necessary.

Secure SDLC Requirements for Vendors

Vendors providing components or services for products in scope of ISO 27001, SOC2, NIST SSDF are expected to comply with the requirements outlined below from the Secure SDLC Framework. Compliance is subject to periodic security audits, third-party assessments, and obligations of each party under the executed contracts.

All vendors are required to provide provenance and integrity information, along with supporting documentation, as defined in the Release Integrity section.

Product component and library vendors must establish development environments aligned with our practices as described in the Secure Development Environment section. They must apply security testing to their components and libraries, as described in the Application Security Testing and Verification section.

Pipeline component vendors must also establish development environments aligned with our practices as described in the Secure Development Environment section. Additionally, their development processes must align with OPSWAT's Secure SDLC Process.

Service vendors are expected to utilize U.S.-based environments that offer a security posture comparable to OPSWAT's services. Their Secure SDLC must include both a Secure SDLC Program and a Secure SDLC Process that mirror OPSWAT's expectations.

Customers' Benefits of Secure SDLC

OPSWAT's Secure SDLC Framework is fully compliant with regulatory requirements and industrial best-practices, ensuring a secure, reliable, and transparent development process.

As a leader in Critical Infrastructure Protection, OPSWAT is committed to achieving the highest level of maturity in Secure SDLC and application security to provide our customers with the following benefits:

- More secure software products, which will minimize exploitation and vulnerabilities.
- Reduction of the risk associated with security breaches and loss of reputation.
- Help address compliance of customer corporate security policies.

GET STARTED

Are you ready to put OPSWAT's solution on the front lines of your cybersecurity strategy?

Talk to one of our experts today.

Scan the QR code or visit us at:
opswat.com/get-started
sales@opswat.com

OPSWAT.
Protecting the World's Critical Infrastructure

For the last 20 years OPSWAT, a global leader in IT, OT, and ICS critical infrastructure cybersecurity, has continuously evolved an end-to-end solutions platform that gives public and private sector organizations and enterprises spanning Financial Services, Defense, Manufacturing, Energy, Aerospace, and Transportation Systems the critical advantage needed to protect their complex networks from cyberthreats.

Built on a "Trust no file. Trust no device.™" philosophy, OPSWAT solves customers' challenges like hardware scanning to secure the transfer of data, files, and device access with zero-trust solutions and patented technologies across every level of their infrastructure. OPSWAT is trusted globally by more than 1,800 organizations, governments, and institutions across critical infrastructure to help secure their devices, files, and networks from known and unknown threats, zero-day attacks, and malware, while ensuring compliance with industry and government-driven policies and regulations.

Discover how OPSWAT is protecting the world's critical infrastructure and securing our way of life.
Visit: www.opswat.com