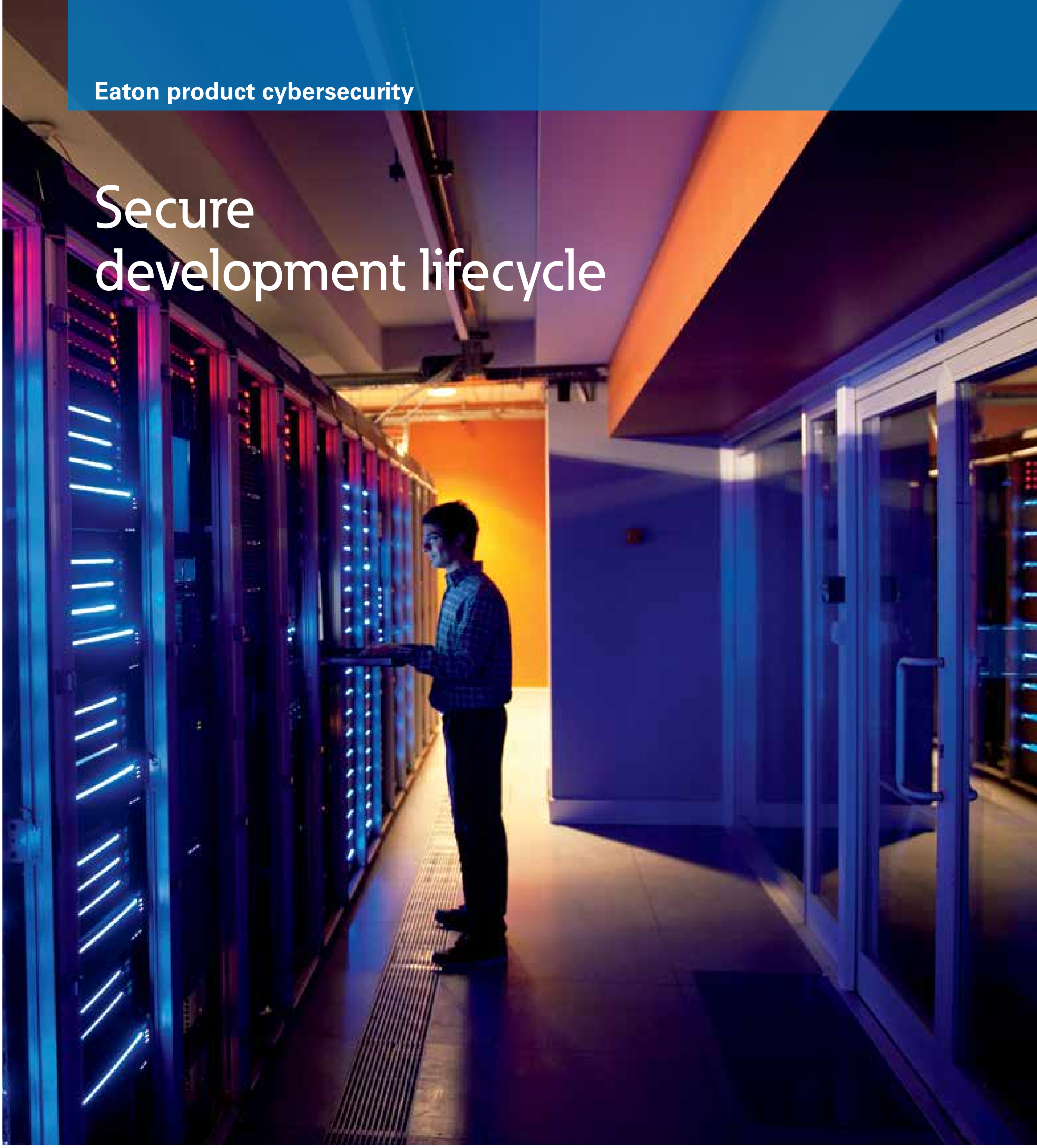
Eaton product cybersecurity

Secure development lifecycle



EATON

Powering Business Worldwide

1. Overview

The secure development lifecycle establishes guidelines for all of Eaton's products and development teams. The activities and the associated deliverables aid in the delivery of a safe, secure, reliable and quality product.

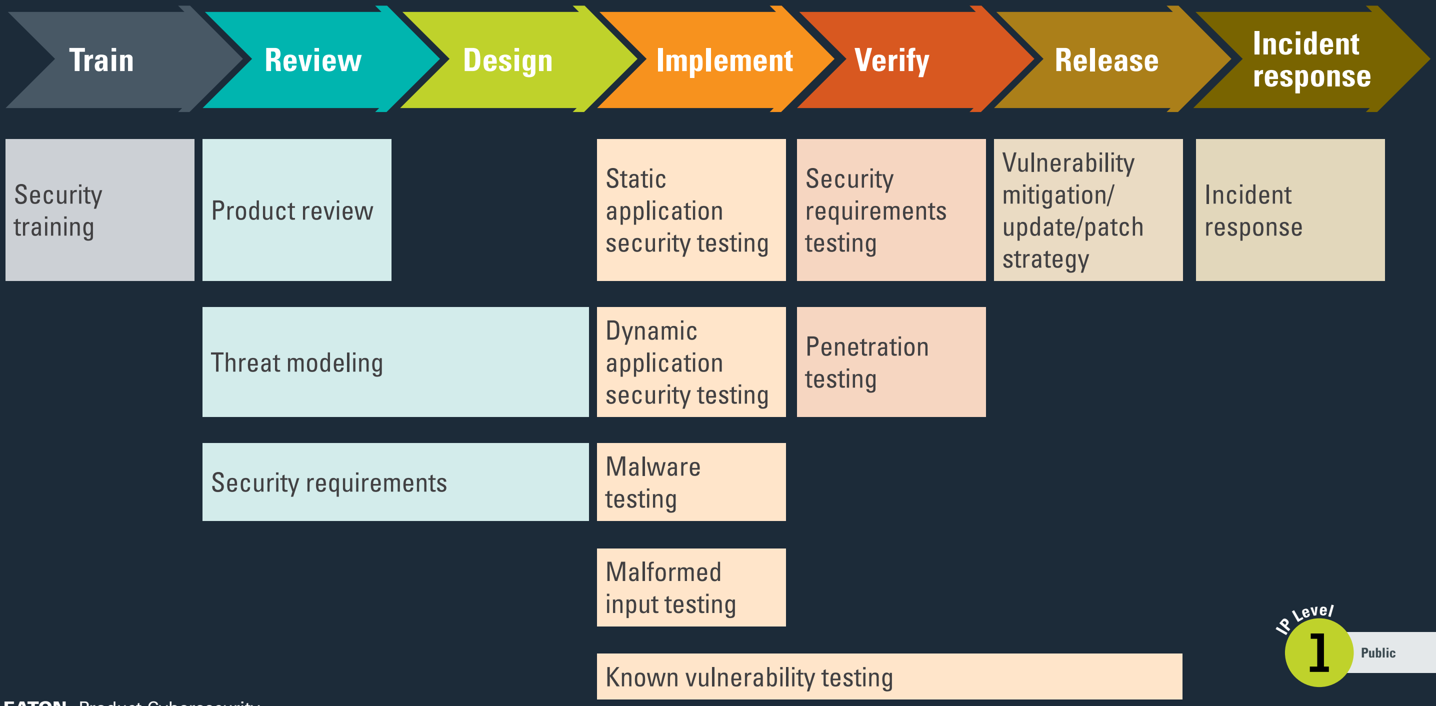
3. Train

All employees are required to complete cybersecurity awareness training. This helps establish an understanding of cybersecurity that can then be used to build more secure products and solutions.

Security training

Developers, testers and managers involved in the development of software or firmware solutions are required to complete additional security training courses. Additional courses cover specific topics such as threat modeling, security requirements, secure coding, and testing.

2. Secure development lifecycle



Train	Review	Design	Implement	Verify	Release	Incident response
Security training	Product review		Static application security testing	Security requirements testing	Vulnerability mitigation/ update/patch strategy	Incident response
	Threat modeling		Dynamic application security testing	Penetration testing		
	Security requirements		Malware testing			
			Malformed input testing			
			Known vulnerability testing			

IP Level 1 Public

4. Review & Design

The product review helps establish security best practices and recommendations while the product is in its infancy. After this review, the product completes a threat modeling and security requirements analysis. Identifying issues in the design reduces the chances of finding flaws in later stages. Additionally, the cost to fix in an early stage is significantly less.

Product review

A product review is used to walk through the concept of the product. Based on the review, a data flow diagram is created to depict the overall flow of data in the product, and an architectural analysis is performed to identify criticality of components. Additionally, sensitive and personal data are identified to ensure compliance with applicable data protection regulations, including the General Data Protection Regulation (GDPR).

Threat modeling

Threat modeling is used to assess risks related to components identified in the architectural analysis. These risks are used to prioritize security requirements and additional mitigations for the product. Identifying threats at an early stage allows design decisions to be made that eliminate or reduce attack surfaces.

Security requirements

Security requirements for the product are identified based on industry standards NIST SP 800-53, NIST SP 800-82, FIPS 140-2, NIST SP 800-124, DHS, IEC 62351, and UL 2900. These requirements provide a unified way of developing a product compliant with multiple security standards.

5. Implement

During the implementation stage, the product is developed and goes through a battery of automated test suites. These suites help identify weaknesses as early as possible.

Static application security testing

Static application security testing (SAST) tools are designed to analyze source code and/or compiled versions of code to help find security flaws. Generally, these tools are automated and produce an output of items. The items aid an analyst in determining the security-relevant portions of code. They can focus efforts on these areas to find flaws.

Dynamic application security testing

Dynamic application security testing (DAST) tools are designed to analyze running applications to help find security flaws. Generally, DAST tools run against web applications, but they could be mobile applications as well. DAST tools look for security vulnerabilities such as cross-site scripting, SQL injection, command injection, path traversal and insecure configurations.

Malware testing

Malware testing is the process of scanning the application with a malware detection tool applicable to a target operating system. This is to identify any known malware that may be present.

Malformed input testing

Malformed input testing (aka fuzz testing) is the process of sending invalid, unexpected or random data to the external interfaces of the product while the product is being monitored. Any unexpected behavior such as a crash, hanging process, assertions, denial of service (DoS), or memory leaks can then be identified. This aids in finding bugs that are generally missed during code reviews.

Known vulnerability testing

Third-party code is a common source of vulnerabilities. Known vulnerability testing is the process of identifying and analyzing all third-party code for known vulnerabilities (CVEs). Once identified, these vulnerabilities can be triaged and mitigated prior to release.

6. Verify

During the verification stage, manual testing is performed in addition to the automated testing described in the implementation stage.

Security requirement testing

Testing is performed against each of the applicable security requirements for the product. These tests are required to confirm compliance with the associated industry standard.

Penetration testing

Penetration testing is a final validation that the code has met all security considerations before going into production. This testing provides a chance to investigate other potentially exploitable avenues for weaknesses.

7. Release

Vulnerability mitigation/update/patch strategy

A vulnerability mitigation/update/patch strategy is defined prior to release. This strategy defines how often updates will occur. Additionally, it defines an incident response plan for new vulnerabilities.

8. Incident response

This is the execution of the incident response plan formulated in the vulnerability mitigation/update/patch strategy.

Notifications of vulnerabilities addressed on our products are posted at our cybersecurity website. Customers are advised to sign up to receive these notifications or report any cybersecurity vulnerabilities on our products.

Powering Business Worldwide

Eaton
1000 Eaton Boulevard
Cleveland, OH 44122
United States
Eaton.com

Follow us on social media to get the
latest product and support information.