




### Functional vs. Security Activities in the SDLC



Software Development Lifecycle	Functional Activities	Security Activities
Requirement	<ul style="list-style-type: none"> <li>Functional requirements</li> <li>Non-functional requirements</li> <li>Technology requirements</li> </ul>	Defining the security requirements
Design	Decide the guidelines and architectural design of project	<ul style="list-style-type: none"> <li>Create a secure design</li> <li>Set secure coding standards</li> <li>Perform threat modeling</li> <li>Secure the architecture</li> </ul>
Development	Functional programming logic Unit testing	<ul style="list-style-type: none"> <li>Implementing security requirements</li> <li>Implementing secure coding standards</li> <li>Adopting secure coding practices</li> </ul>
Testing	Functional testing such as black-, grey-, and white-box testing	Security testing
Deployment	Deployment	Ensure secure deployment
Maintenance	Update functionality	Update the system with security patches

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

**Notes:** \_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

## Advantages of Integrating Security in the SDLC



- Reduces the presence of **software vulnerabilities** to a great extent
- Can **comply with the regulations**, standards, or requirements for secure software development
- Reduce **costly rework** by detecting and eliminating flaws at the earliest phase
- Improves developer **job satisfaction**
- Improves **customer satisfaction**
- Embeds **security culture** to improve quality and reliability
- Reuses **trusted software** in future development
- Reduces **maintenance costs**

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

## Security Requirements



- Non-functional** requirements that need to be addressed to maintain the **confidentiality, integrity, and availability** of the application
- Stakeholders** often overlook security requirement during the inception phase of software development
- This **negligence** may result in the application being vulnerable to different types of attacks or abuse
- Gathering security requirements should be part of the **strategic application development process**

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_