KaozhengPro

IT認證考試題庫

專業平臺

考證寶提供最新考古題與模擬試題
協助您高效通過認證考試

www.kaozhengpro.com

Exam : **312-97**

Title : EC-Council Certified DevSecOps Engineer (ECDE)

Version : DEMO

1.Scenario: You are tasked with reviewing the security posture of a project that uses multiple open-source libraries.
What is the first step you should take to assess the security risks of these libraries?
A. Perform an automated vulnerability scan of the libraries.
B. Run static code analysis on the libraries' source code.
C. Rely on community feedback and star ratings to assess library security.
D. Review the open-source licenses to ensure compliance.
Answer: A
Explanation:
Performing an automated vulnerability scan of the libraries allows you to assess the security risks associated with using those open-source components. It ensures that libraries with known vulnerabilities are flagged and addressed before they pose a threat.

2.During a compliance audit, it was found that several environments do not meet HIPAA's requirements for secure data storage.
What should be the next step to ensure compliance across the development pipeline?
A. Restrict compliance checks to the production environment only to ensure smooth operation.
B. Implement end-to-end encryption for all environments handling sensitive data.
C. Perform post-deployment reviews once per quarter to identify and address compliance gaps.
D. Delay compliance checks until the final pre-production stage to minimize delays.
Answer: B
Explanation:
End-to-end encryption is the most effective way to secure data across all environments, meeting HIPAA's strict requirements for protecting sensitive information. By encrypting data throughout the entire pipeline, the organization can ensure compliance, even if there are vulnerabilities in other areas of the system.

3.Scenario: Your organization is developing a web-based application that will handle sensitive data.
How can you ensure that security is incorporated into the design and development phases?
A. Focus on end-user security awareness training during the rollout phase.
B. Rely on manual code reviews to ensure secure coding practices.
C. Schedule a security audit once the application is near completion.
D. Integrate static code analysis and threat modeling into the early stages of development.
Answer: D
Explanation:
Integrating static code analysis and threat modeling into the early stages of development helps detect security issues in the code before it progresses to later stages. This proactive approach ensures that security is built into the design and development process, reducing the risk of vulnerabilities in production.

4.During the development of a cloud-based application, which practice should a team adopt to ensure comprehensive threat modeling?
A. Conducting a manual review of security controls once per development sprint.
B. Integrating automated threat modeling tools into the CI/CD pipeline for continuous threat assessment.
C. Having quarterly third-party security audits to validate the threat model's effectiveness.

D. Hosting monthly security workshops to discuss and update the threat model with new findings.
Answer: B
Explanation:
Integrating automated threat modeling tools into the CI/CD pipeline allows for continuous threat assessment, which is vital in the fast-paced development environments of cloud-based applications. This practice ensures that any changes in the application or its environment are assessed in real-time, providing ongoing assurance that the application's security posture is maintained throughout its development lifecycle.

5.What is a common security issue in traditional DevOps workflows that involves inadequate handling of security alerts?
A. Inadequate monitoring of network traffic to detect anomalous activities that could indicate a breach.
B. Overlooking the encryption of sensitive data at rest and in transit within the CI/CD pipeline.
C. Not enforcing strong access control measures on production servers and development environments.
D. Failing to prioritize and address high-severity security alerts in a timely manner.
Answer: D
Explanation:
In traditional DevOps workflows, failing to prioritize and address high-severity security alerts can lead to unmitigated risks and potential breaches. This issue stems from the reactive, rather than proactive, approach to security, which can result in significant vulnerabilities if critical alerts are delayed or ignored.