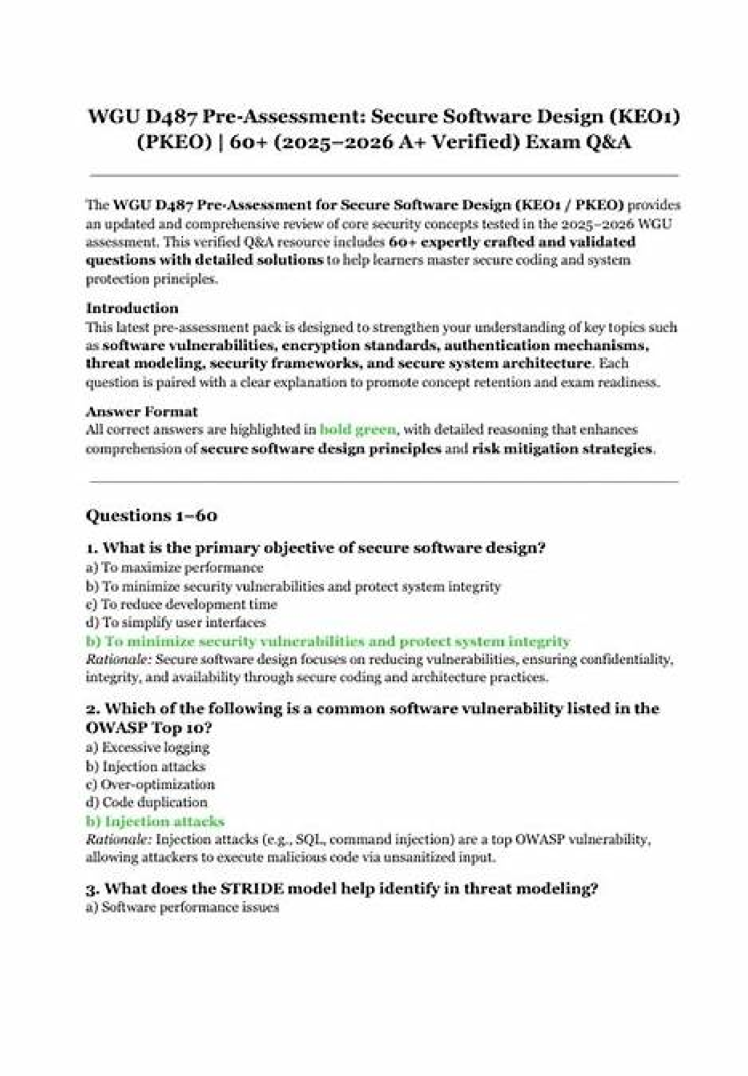
# High Hit Rate WGUSecure Software Design (KEO1) Exam Test Torrent Has a High Probability to Pass the Exam



## WGU D487 Pre-Assessment: Secure Software Design (KEO1) (PKEO) | 60+ (2025–2026 A+ Verified) Exam Q&A

The **WGU D487 Pre-Assessment for Secure Software Design (KEO1 / PKEO)** provides an updated and comprehensive review of core security concepts tested in the 2025–2026 WGU assessment. This verified Q&A resource includes **60+ expertly crafted and validated questions with detailed solutions** to help learners master secure coding and system protection principles.

**Introduction**
This latest pre-assessment pack is designed to strengthen your understanding of key topics such as **software vulnerabilities, encryption standards, authentication mechanisms, threat modeling, security frameworks, and secure system architecture**. Each question is paired with a clear explanation to promote concept retention and exam readiness.

**Answer Format**
All correct answers are highlighted in **bold green**, with detailed reasoning that enhances comprehension of **secure software design principles** and **risk mitigation strategies**.

### Questions 1–60

**1. What is the primary objective of secure software design?**
a) To maximize performance
b) To minimize security vulnerabilities and protect system integrity
c) To reduce development time
d) To simplify user interfaces
**b) To minimize security vulnerabilities and protect system integrity**
*Rationale:* Secure software design focuses on reducing vulnerabilities, ensuring confidentiality, integrity, and availability through secure coding and architecture practices.

**2. Which of the following is a common software vulnerability listed in the OWASP Top 10?**
a) Excessive logging
b) Injection attacks
c) Over-optimization
d) Code duplication
**b) Injection attacks**
*Rationale:* Injection attacks (e.g., SQL, command injection) are a top OWASP vulnerability, allowing attackers to execute malicious code via unsanitized input.

**3. What does the STRIDE model help identify in threat modeling?**
a) Software performance issues

Will you feel that the product you have brought is not suitable for you? One trait of our Secure-Software-Design exam prepare is that you can freely download a demo to have a try. Because there are excellent free trial services provided by our Secure-Software-Design exam guides, our products will provide three demos that specially designed to help you pick the one you are satisfied. On the one hand, by the free trial services you can get close contact with our products, learn about the detailed information of our Secure-Software-Design Study Materials, and know how to choose the right version of our Secure-Software-Design exam questions.

## WGU Secure-Software-Design Exam Syllabus Topics:

| Topic | Details |
|---|---|
| Topic 1 | - Software Architecture and Design: This module covers topics in designing, analyzing, and managing large scale software systems. Students will learn various architecture types, how to select and implement appropriate design patterns, and how to build well structured, reliable, and secure software systems. |

| | |
|---|---|
| Topic 2 | - Software System Management: This section of the exam measures skills of Software Project Managers and covers the management of large scale software systems. Learners study approaches for overseeing software projects from conception through deployment. The material focuses on coordination strategies and management techniques that ensure successful delivery of complex software solutions. |
| Topic 3 | - Large Scale Software System Design: This section of the exam measures skills of Software Architects and covers the design and analysis of large scale software systems. Learners investigate methods for planning complex software architectures that can scale and adapt to changing requirements. The content addresses techniques for creating system designs that accommodate growth and handle increased workload demands. |

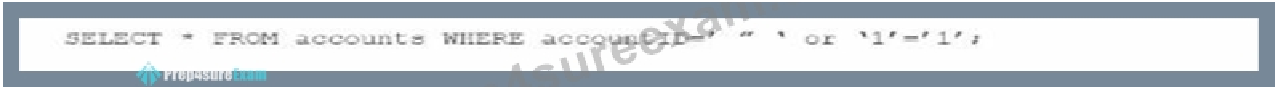
**>> Learning Secure-Software-Design Materials <<**

## Secure-Software-Design Reliable Test Camp - Pdf Secure-Software-Design Files

As one of the most professional dealer of practice materials, we have connection with all academic institutions in this line with proficient researchers of the knowledge related with the Secure-Software-Design Practice Exam to meet your tastes and needs, please feel free to choose. We want to specify all details of various versions. You can decide which one you prefer, when you made your decision and we believe your flaws will be amended and bring you favorable results even create chances with exact and accurate content.

## WGUSecure Software Design (KEO1) Exam Sample Questions (Q10-Q15):

**NEW QUESTION # 10**
An individual is developing a software application that has a back-end database and is concerned that a malicious user may run the following SOL query to pull information about all accounts from the database:



```
SELECT * FROM accounts WHERE accountID=' " ' or '1'='1';
```

Which technique should be used to detect this vulnerability without running the source codes?

- A. Dynamic analysis
- B. Fuzz testing
- C. Cross-site scripting
- D. Static analysis

**Answer: D**

Explanation:
Static analysis is a method used to detect vulnerabilities in software without executing the code. It involves examining the codebase for patterns that are indicative of security issues, such as SQL injection vulnerabilities. This technique can identify potential threats and weaknesses by analyzing the code's structure, syntax, and data flow.
:
Static analysis as a means to identify security vulnerabilities1.
The importance of static analysis in the early stages of the SDLC to prevent security issues2.
Learning-based approaches to fix SQL injection vulnerabilities using static analysis3.

**NEW QUESTION # 11**
The product team has been tasked with updating the user interface (UI). They will change the layout and also add restrictions to field lengths and what data will be accepted.
Which secure coding practice is this?

- A. Data protection
- B. Input validation
- C. Communication security

- D. Access control

**Answer: B**

Explanation:
Comprehensive and Detailed Explanation From Exact Extract:
This is an example of Input validation, which involves ensuring all user inputs conform to expected formats, lengths, and content before processing. Restricting field lengths and validating accepted data types prevents injection attacks, buffer overflows, and improper data handling. Access control (B) restricts user permissions, communication security (C) protects data in transit, and data protection (D) focuses on confidentiality and integrity of stored data. OWASP Secure Coding Practices and Microsoft SDL emphasize rigorous input validation as a first line of defense against many vulnerabilities.
References:
OWASP Secure Coding Practices - Input Validation
Microsoft SDL Secure Coding Guidelines
NIST SP 800-53: Security and Privacy Controls for Information Systems

**NEW QUESTION # 12**
Which software development model starts by specifying and implementing just a part of the software, which is then reviewed and identifies further requirements that are implemented by repeating the cycle?

- A. Iterative
- B. Implementation
- C. Code and fix
- D. Waterfall

**Answer: A**

Explanation:
Comprehensive and Detailed Explanation From Exact Extract:
The Iterative software development model fits this description. It involves specifying and implementing a portion of the software, reviewing it, gathering feedback, and refining or adding requirements in successive cycles. This approach supports evolving requirements and continuous improvement. Iterative models contrast with Waterfall (C), which is linear and sequential, with no repetition of phases. "Code and fix" (D) is an informal, ad hoc process lacking formal review cycles. Implementation (B) is a phase, not a model. The iterative approach is advocated in ISO/IEC 12207 and NIST guidelines for secure development, as it allows early detection and remediation of security issues by incremental design and testing.
References:
ISO/IEC 12207 Software Lifecycle Processes
NIST SP 800-64 Revision 2: Security Considerations in SDLC
Microsoft SDL Documentation

**NEW QUESTION # 13**
A potential threat was discovered during vulnerability testing when an environment configuration file was found that contained the database username and password stored in plain text.
How should existing security controls be adjusted to prevent this in the future?

- A. Ensure Strong Password Policies are in Effect
- B. Encrypt Secrets in Storage and Transit
- C. Enforce Role-Based Authorization
- D. Validate All User Input

**Answer: B**

**NEW QUESTION # 14**
The security team is identifying technical resources that will be needed to perform the final product security review.
Which step of the final product security review process are they in?

- A. Evaluate and Plan for Remediation
- B. Release and Ship

- C. Assess Resource Availability
- D. Identify Feature Eligibility

**Answer: C**

**NEW QUESTION # 15**

......

We update the Secure-Software-Design study materials frequently to let the client practice more and follow the change of development in the practice and theory. So that our worthy customers can always receive the most updated and the latest Secure-Software-Design learning guide. And according to our service, you can enjoy free updates for one year after you pay for the Secure-Software-Design Exam Questions. So if we update it, then we will auto send it to you. You won't miss any information that you need to pass the exam.