

Formulário Pré-teste de Conhecimento

As questões a seguir têm o objetivo de avaliar seu conhecimento sobre segurança de software e padrões de segurança após a sua participação na atividade de ensino e aprendizagem utilizando uma aula tradicional e o RiskGuard. Para isso, siga as recomendações abaixo:

- Responda com base exclusivamente em seu conhecimento adquirido, sem consultar materiais, colegas ou outras fontes externas.
- Busque responder corretamente ao maior número possível de questões. No entanto, caso não saiba a resposta, marque a opção "Não sei" em vez de escolher uma alternativa aleatória.

Observações: Antes de enviar o formulário, revise suas respostas para garantir que não deixou nenhum campo em branco por engano.

1. Qual o seu nome? (Obrigatória)

2. Qual o seu e-mail? (Obrigatória)

3. Como funciona um ataque de Injection? (Obrigatória)

() Um ataque de injection pode ser prevenido através da criação de códigos-fonte obfuscados, o que dificulta a leitura dos scripts por atacantes e impede qualquer tentativa de injeção de comandos no sistema.

() Em um ataque de injection, um atacante insere comandos maliciosos em um sistema vulnerável. Uma senha forte e complexa pode impedir que isso aconteça, pois ela adiciona uma camada de segurança extra, dificultando a execução de injeções.

() Um ataque de injection ocorre quando um sistema permite que dados maliciosos sejam injetados em entradas não validadas, como formulários. Sistemas que validam e sanitizam entradas corretamente podem impedir que esses ataques sejam bem-sucedidos, protegendo a integridade dos dados e do sistema.

() Ataques de injection podem ser prevenidos apenas por ferramentas antivírus que detectam código malicioso. Essas ferramentas bloqueiam automaticamente qualquer tentativa de injeção de dados no sistema.

() Não sei

4. Quais são as consequências de uma falha criptográfica em uma aplicação web? (Obrigatória)

() Falha de segurança que pode ser corrigida sem comprometer os dados.

- Aumento na confiança dos usuários devido à criptografia eficaz.
- Melhora no desempenho do sistema devido ao uso de algoritmos mais rápidos.
- Vazamento de dados sensíveis, como senhas e informações pessoais.
- Não sei

5. Relacione corretamente as vulnerabilidades com os padrões de segurança correspondentes, escolhendo a alternativa correta: (Obrigatória)

Vulnerabilidade	Padrão de Segurança
A. Injeção de SQL	1. Validação de Entrada
B. Cross-Site Scripting (XSS)	2. Codificação de Saída
C. Autenticação Quebrada	3. Autenticação Multifator
D. Referência Direta Insegura a Objetos	4. Aplicação de Controle de Acesso

- 1 → B, 2 → A, 3 → D, 4 → C
- 1 → A, 2 → B, 3 → C, 4 → D
- 1 → D, 2 → C, 3 → A, 4 → B
- 1 → C, 2 → D, 3 → B, 4 → A
- Não sei

6. Como se prevenir de um ataque de Violão/Quebra do Controle de Acesso? (Obrigatória)

- Usar apenas URLs para determinar quem pode acessar o quê no sistema.
- Permitir que qualquer pessoa acesse o sistema sem restrições.
- Implementação de Políticas de Privilégio Mínimo e Access Control.
- Permitir que os usuários modifiquem suas próprias permissões de acesso.
- Não sei

7. Sobre injeção de SQL e segurança em bancos de dados, responda: Um sistema web armazena credenciais de usuários em um banco de dados. Qual das seguintes abordagens ainda pode ser vulnerável a um ataque de injeção de SQL, mesmo que pareça segura? (Obrigatória)

- Uso de stored procedures que não concatenam entradas do usuário diretamente em comandos SQL.
- Validação de entrada do usuário apenas no frontend antes de enviar os dados ao servidor.

() Utilização de ORM (Object-Relational Mapping) com suporte nativo a escaping de entradas.

() Uso de consultas parametrizadas (prepared statements) com a verificação de tipos das entradas do usuário.

() Não sei

8. Por que o registro e o monitoramento de atividades são fundamentais para a segurança de um sistema? (Obrigatória)

() São úteis apenas para fins de auditoria fiscal e não afetam a segurança.

() Substituem a necessidade de autenticação multifator (MFA).

() Permitem identificar atividades suspeitas e responder rapidamente a incidentes.

() Eliminam completamente a necessidade de aplicar patches de segurança.

() Não sei

9. Sobre falhas criptográficas e armazenamento seguro de credenciais, responda: Em relação ao armazenamento de senhas, qual das abordagens abaixo representa a melhor prática em termos de segurança? (Obrigatória)

() Criar um algoritmo de hash próprio para senhas, pois soluções conhecidas podem ser comprometidas.

() Armazenar as senhas em texto plano, mas protegidas por um firewall e acesso restrito ao banco de dados.

() Usar um algoritmo de hash forte, como bcrypt, Argon2 ou PBKDF2, com um salt único para cada senha.

() Armazenar senhas criptografadas com AES-256 e uma chave fixa embutida no código-fonte.

() Usar a mesma chave de criptografia em vários sistemas diferentes.

() Não sei

10. Sobre quebra de controle de acesso. Qual das seguintes configurações pode levar a uma vulnerabilidade de quebra de controle de acesso em um sistema web? (Obrigatória)

() Implementação de listas de controle de acesso (ACLs) para definir permissões por função.

() Implementação de controle de acesso no frontend sem reforço no backend.

() Aplicação do princípio do mínimo privilégio em todas as funções administrativas.

() Uso de JSON Web Tokens (JWT) com um tempo de expiração curto.

Não sei

11. Após um ataque em uma plataforma de e-commerce, onde dados de pagamento dos clientes foram alterados por um atacante, qual seria a melhor prática para evitar falhas de integridade de dados? (Obrigatória)

Permitir que os clientes alterem suas próprias transações diretamente pelo sistema, sem controles adicionais.

Ignorar a validação de integridade dos dados ao receber dados de terceiros, como processadores de pagamento.

Implementar criptografia robusta de dados sensíveis, como números de cartão de crédito, e usar autenticação multifatorial para acesso administrativo.

Não permitir que os dados de pagamento sejam criptografados, para que os funcionários possam acessá-los facilmente.

Não sei

12. Em uma empresa, um sistema de controle de versões foi comprometido devido à falta de verificação da autenticidade do código. O que poderia ser feito para melhorar a integridade do software e proteger contra modificações não autorizadas? (Obrigatória)

Armazenar o código-fonte em servidores públicos para garantir acesso fácil e rápido.

Não aplicar patches ou atualizações no código-fonte para evitar riscos de falhas de integridade.

Implementar assinaturas digitais no código-fonte e usar um sistema de controle de versões que registre todas as alterações, com auditoria rigorosa.

Permitir que qualquer funcionário faça modificações no código-fonte, sem controle ou aprovação.

Não sei

13. Em um site de e-commerce, um usuário altera o ID na URL de sua conta de "12345" para "12346" e acessa os dados de outro usuário. Qual vulnerabilidade de segurança está ocorrendo? (Obrigatória)

CSRF

Broken Access Control

Injeção de SQL.

XSS.

Não sei

14. Um atacante explora uma falha em um sistema de software, alterando o código-fonte ou os dados armazenados para modificar o comportamento do sistema, comprometendo sua precisão e confiabilidade. Qual vulnerabilidade está sendo explorada neste cenário? (Obrigatória)

Manipulação de código ou dados (exploiting software integrity)

Cross-Site Scripting (XSS).

Phishing

Injeção de SQL

Não sei