Desenvolvimento Web Seguro – Notas de Aula

Semana 02 – Autenticação e Autorização

Prof. Jefferson O. Andrade

2024-05-15

Contents

1 Introdução

A autenticação e a autorização são componentes críticos na segurança de aplicações web. Eles garantem que somente usuários legítimos possam acessar o sistema e que esses usuários só possam realizar ações para as quais estão autorizados. Aqui estão algumas razões pelas quais a autenticação e a autorização são essenciais:

1. **Proteção de Dados Sensíveis:**
    - A autenticação garante que apenas usuários autorizados possam acessar informações sensíveis, como dados pessoais, financeiros e comerciais. Sem autenticação adequada, esses dados podem

ser facilmente acessados por indivíduos não autorizados, levando a vazamentos de informações e comprometimento de privacidade.

2. **Prevenção de Acesso Não Autorizado:**
   - A autorização assegura que os usuários, mesmo que autenticados, só possam realizar ações permitidas com base em suas permissões. Isso impede que usuários com permissões limitadas acessem ou modifiquem recursos críticos do sistema, como configurações administrativas ou dados de outros usuários.

3. **Redução de Riscos de Ataques:**
   - Implementações fracas de autenticação e autorização são alvos comuns de ataques cibernéticos, como força bruta, roubo de credenciais e escalonamento de privilégios. Métodos robustos de autenticação, como a autenticação multifator (MFA), e modelos de autorização bem definidos, como RBAC (Role-Based Access Control), ajudam a mitigar esses riscos.

4. **Conformidade com Regulamentações:**
   - Muitas regulamentações de proteção de dados e privacidade, como o GDPR (Regulamento Geral sobre a Proteção de Dados) e a LGPD (Lei Geral de Proteção de Dados), exigem que as organizações implementem medidas adequadas de autenticação e autorização para proteger os dados dos usuários. A conformidade com essas regulamentações é crucial para evitar penalidades legais e manter a confiança dos clientes.

5. **Manutenção da Integridade do Sistema:**
   - A autenticação e a autorização protegem a integridade do sistema ao garantir que apenas usuários legítimos possam realizar ações que afetam o funcionamento e a segurança da aplicação. Isso inclui prevenir que atacantes introduzam códigos maliciosos ou façam alterações não autorizadas no sistema.

A autenticação e a autorização são pilares fundamentais na construção de aplicações web seguras. Elas não apenas protegem os dados e a privacidade dos usuários, mas também garantem a integridade e a confiabilidade do sistema como um todo.

2 Autenticação e Autorização

2.1 Definição de Autenticação:

Autenticação é o processo de verificar a identidade de um usuário ou sistema. Em outras palavras, é a etapa onde o sistema confirma que o usuário é quem ele afirma ser. Isso geralmente é feito por meio de credenciais, como nome de usuário e senha, mas também pode envolver métodos mais avançados, como biometria (impressões digitais, reconhecimento facial) ou autenticação multifator (MFA), que combina vários métodos de verificação.

2.2 Definição de Autorização:

Autorização é o processo de conceder a um usuário ou sistema permissão para acessar recursos específicos ou realizar certas ações dentro de um sistema. Enquanto a autenticação verifica a identidade, a autorização determina o que essa identidade pode fazer. Por exemplo, após um usuário ser autenticado, o sistema verifica suas permissões para acessar arquivos, executar comandos ou modificar dados.

2.3 Diferenças Entre Autenticação e Autorização:

1. **Propósito:**
   - **Autenticação:** Verifica a identidade do usuário. É o processo de garantir que o usuário é quem ele diz ser.
   - **Autorização:** Controla o acesso aos recursos e ações dentro do sistema. Determina o que o usuário autenticado pode fazer.
2. **Etapas do Processo:**
   - **Autenticação:** Ocorre primeiro. Sem autenticação, o sistema não pode determinar quais permissões conceder ao usuário.
   - **Autorização:** Ocorre após a autenticação. Somente usuários autenticados podem ser autorizados a acessar recursos ou executar ações específicas.
3. **Métodos Comuns:**
   - **Autenticação:** Nome de usuário e senha, tokens, biometria, autenticação multifator (MFA).
   - **Autorização:** Controle de Acesso Baseado em Funções (RBAC), Controle de Acesso Baseado em Atributos (ABAC), listas de controle de acesso (ACLs).
4. **Papel na Segurança:**
   - **Autenticação:** Protege o sistema contra acesso não autorizado verificando identidades.
   - **Autorização:** Protege recursos específicos e funcionalidades do sistema, garantindo que somente usuários autorizados possam acessá-los ou executá-los.

2.4 Exemplo Prático:

- **Autenticação:** Quando um usuário faz login em um site com seu nome de usuário e senha, o sistema verifica essas credenciais para confirmar a identidade do usuário.
- **Autorização:** Após o login, o sistema verifica as permissões do usuário. Por exemplo, um usuário comum pode acessar seu próprio perfil e dados, enquanto um administrador pode ter acesso a funções adicionais, como gerenciamento de usuários e configurações do sistema.

A autenticação e a autorização são processos distintos, mas complementares, que juntos garantem que somente usuários legítimos e autorizados possam acessar recursos e executar ações dentro de um sistema.

3 Métodos Comuns de Autenticação

3.1 Nome de Usuário e Senha

- **Descrição:** Este é o método mais comum de autenticação, onde os usuários fornecem um nome de usuário (ou ID) e uma senha para acessar um sistema.
- **Vantagens:** Simples e fácil de implementar.
- **Desvantagens:** Vulnerável a ataques de força bruta, phishing e reutilização de senhas. Requer políticas fortes de criação e gerenciamento de senhas para ser eficaz.

3.2 Tokens de Autenticação

- **Descrição:** Tokens são gerados pelo servidor e enviados ao cliente após a autenticação inicial. Os tokens são então usados para autenticar solicitações subsequentes sem necessidade de enviar novamente o nome de usuário e senha.
- **Tipos Comuns:** JWT (JSON Web Tokens), tokens de sessão.
- **Vantagens:** Reduz a exposição das credenciais do usuário e pode ser usado para autenticação sem estado (stateless authentication).
- **Desvantagens:** Necessita de mecanismos seguros para geração, armazenamento e invalidação de tokens.

3.3 Biometria

- **Descrição:** Usa características físicas ou comportamentais únicas do usuário, como impressões digitais, reconhecimento facial ou reconhecimento de voz, para autenticar a identidade.
- **Vantagens:** Difícil de falsificar, elimina a necessidade de memorizar senhas.
- **Desvantagens:** Pode ser invasivo em termos de privacidade, requer hardware especializado, e pode ser vulnerável a ataques sofisticados.

3.4 Autenticação Multifator (MFA)

- **Descrição:** Combina dois ou mais métodos de autenticação, como algo que o usuário conhece (senha), algo que o usuário possui (token), e algo que o usuário é (biometria).
- **Vantagens:** Significativamente mais seguro que métodos de autenticação únicos, pois um invasor teria que comprometer múltiplos fatores para obter acesso.
- **Desvantagens:** Pode ser mais complexo e caro de implementar, pode causar inconveniência ao usuário.

3.5 Autenticação Baseada em Certificados

- **Descrição:** Usa certificados digitais para autenticar a identidade do usuário ou dispositivo. Os certificados são emitidos por uma Autoridade Certificadora (CA) confiável.
- **Vantagens:** Alta segurança, pode ser usado para autenticação de dispositivos além de usuários.
- **Desvantagens:** Requer uma infraestrutura de gerenciamento de certificados (PKI), pode ser complexo de administrar.

3.6 Autenticação OAuth/OpenID Connect

- **Descrição:** Permite que os usuários autentiquem-se usando contas de terceiros (como Google, Facebook) sem precisar criar novas credenciais.
- **Vantagens:** Conveniente para usuários, reduz a necessidade de gerenciar múltiplas senhas.
- **Desvantagens:** Depende da segurança e disponibilidade do provedor de identidade terceiro.

4 Mecanismos de Autorização

4.1 Controle de Acesso Baseado em Funções (RBAC)

4.1.1 Definição

O Controle de Acesso Baseado em Funções (Role-Based Access Control - RBAC) é um modelo de controle de acesso onde as permissões são atribuídas a funções específicas dentro de uma organização, e os usuários são então atribuídos a essas funções. Cada função tem um conjunto definido de permissões que determinam o que os usuários naquela função podem acessar e realizar dentro do sistema.

4.1.2 Características Principais

- **Funções:** Representam um conjunto de permissões associadas a uma função específica dentro da organização (por exemplo, administrador, editor, visualizador).
- **Permissões:** São as ações que podem ser realizadas dentro do sistema (por exemplo, ler, escrever, excluir).
- **Usuários:** São atribuídos a uma ou mais funções, herdando as permissões associadas a essas funções.

4.1.3 Vantagens

- **Facilidade de Gerenciamento:** Simplifica a administração de permissões, pois as permissões são atribuídas a funções e não diretamente a usuários.
- **Escalabilidade:** Facilita a adição de novos usuários e a atribuição de permissões à medida que a organização cresce.
- **Segurança:** Reduz a possibilidade de erro humano na atribuição de permissões, garantindo que os usuários tenham apenas as permissões necessárias para suas funções.

4.1.4 Desvantagens

- **Rigidez:** Pode ser inflexível em cenários onde os requisitos de acesso são dinâmicos e mudam frequentemente.
- **Manutenção:** Requer manutenção constante para garantir que as funções e suas permissões estejam atualizadas e alinhadas com as necessidades da organização.

4.2 Controle de Acesso Baseado em Atributos (ABAC)

4.2.1 Definição

O Controle de Acesso Baseado em Atributos (Attribute-Based Access Control - ABAC) é um modelo de controle de acesso onde as permissões são atribuídas com base em um conjunto de atributos e políticas que determinam as condições sob as quais o acesso é permitido. Os atributos podem incluir características do usuário, do recurso ou do ambiente.

4.2.2 Características Principais

- **Atributos:** Podem incluir atributos do usuário (por exemplo, função, departamento), atributos do recurso (por exemplo, tipo de documento, classificação), e atributos do ambiente (por exemplo, horário, localização).
- **Políticas:** Definem as regras que determinam as condições sob as quais o acesso é permitido ou negado. As políticas são escritas em termos de atributos e suas combinações.

- **Decisões de Acesso:** São feitas em tempo real com base na avaliação dos atributos e nas políticas definidas.

4.2.3 Vantagens

- **Flexibilidade:** Oferece um controle de acesso mais granular e dinâmico, adaptando-se facilmente a mudanças nos requisitos de acesso.
- **Segurança:** Permite a implementação de políticas de segurança complexas que consideram múltiplos fatores, melhorando a precisão das decisões de acesso.
- **Personalização:** Permite personalizar o acesso com base em uma ampla variedade de atributos, proporcionando um controle mais detalhado.

4.2.4 Desvantagens

- **Complexidade:** Pode ser mais complexo de implementar e gerenciar devido à necessidade de definir e manter políticas e atributos.
- **Desempenho:** A avaliação em tempo real de atributos e políticas pode impactar o desempenho, especialmente em sistemas com grande volume de acessos.

5 Estudos de Caso

5.1 Caso Target (2013)

- **Descrição:** Em 2013, a Target sofreu uma das maiores violações de dados da história, resultando no comprometimento de informações de cartão de crédito e débito de aproximadamente 40 milhões de clientes.
- **Problema:** Os atacantes exploraram credenciais de um fornecedor de HVAC (aquecimento, ventilação e ar condicionado) para acessar a rede da Target. Uma vez dentro, eles conseguiram se mover lateralmente pela rede devido a falhas de autorização inadequadas.
- **Consequências:** Perda financeira significativa, dano à reputação e custos legais elevados.
- **Lições Aprendidas:** Implementação de autenticação multifator (MFA) e segmentação de rede para limitar o acesso a partes críticas do sistema.

5.2 Caso Uber (2016)

- **Descrição:** Em 2016, a Uber sofreu uma violação de dados que expôs informações pessoais de 57 milhões de motoristas e passageiros.
- **Problema:** Os atacantes conseguiram acessar uma base de dados armazenada na AWS usando credenciais de acesso que foram acidentalmente expostas em um repositório de código no GitHub.
- **Consequências:** Pagamento de US$ 148 milhões em um acordo judicial, além de danos à reputação.
- **Lições Aprendidas:** Uso de práticas seguras de gerenciamento de credenciais e autenticação multifator para acessar recursos sensíveis.

5.3 Caso Equifax (2017)

- **Descrição:** Em 2017, a Equifax sofreu uma violação massiva de dados que expôs informações pessoais de 147 milhões de pessoas.
- **Problema:** A vulnerabilidade inicial foi uma falha de autorização em um portal web da Equifax. A falta de segmentação adequada e de controles de autorização permitiu que os atacantes acessassem grandes volumes de dados.
- **Consequências:** Multas significativas, perda de confiança do consumidor e altos custos de mitigação e reparação.
- **Lições Aprendidas:** Implementação de controles de autorização robustos e segmentação de rede para limitar o acesso a dados sensíveis.

5.4 Caso Facebook (2018)

- **Descrição:** Em 2018, o Facebook revelou uma violação que afetou cerca de 50 milhões de contas de usuários.
- **Problema:** Uma falha no sistema de autenticação permitiu que atacantes roubassem tokens de acesso, que poderiam ser usados para assumir o controle das contas dos usuários.
- **Consequências:** Exposição de informações pessoais, investigações regulatórias e danos à reputação.
- **Lições Aprendidas:** Revisão e fortalecimento dos mecanismos de autenticação e implementação de monitoramento contínuo para detectar atividades suspeitas.

5.5 Caso Twitter (2020)

- **Descrição:** Em 2020, várias contas de alto perfil no Twitter foram comprometidas em um ataque de engenharia social.
- **Problema:** Os atacantes enganaram os funcionários do Twitter para que revelassem suas credenciais, permitindo acesso às ferramentas internas de administração.
- **Consequências:** Controle temporário de contas de alto perfil, divulgação de mensagens fraudulentas e investigação por agências governamentais.
- **Lições Aprendidas:** Educação contínua dos funcionários sobre engenharia social e implementação de autenticação multifator para acessos administrativos.