Papéis de acesso

Conceda ações de acordo com a função

Pré-requisitos

Conheça a diferença entre autenticação e autorização e os papéis levantados na descoberta. O exercício verifica a parte funcional de uma política com Node.js e dados fictícios. O escopo de unidade e o estado da reserva são verificações adicionais do serviço real.

Separe operação, permissão e atribuição [1]

Uma operação é algo que o sistema executa, como registrar check-in. Uma permissão autoriza uma operação definida pela política. O papel agrupa permissões adequadas a uma função. A atribuição liga o papel à conta. Assim, para remover o acesso de uma pessoa que saiu da recepção, retire a atribuição dela sem mudar as permissões da equipe.

Conceda o mínimo necessário [2]

Quem atende precisa consultar a agenda e registrar chegadas. A gestão edita a disponibilidade, e o suporte consulta incidentes operacionais. Cada papel deve ter uma lista explícita de ações. O título de gestor, por si só, não exige a capacidade de fazer check-in. Se a pessoa acumular as duas funções, a política deste exercício permite a união das atribuições ativas.

Reavalie a política no ponto de execução

O servidor obtém os papéis de uma origem confiável e verifica a ação a cada solicitação. A interface pode esconder uma ação indisponível para orientar o usuário. Antes de tocar a reserva, o serviço verifica também o recurso e o escopo. Se a função for revogada, as decisões futuras devem refletir a revogação conforme a política de sessão.

Aplique recusa por padrão [2]

Se uma nova operação como estorno for implantada sem concessão explícita, a matriz inicial a recusa. Uma lista de exceções que só enumera operações proibidas tende a liberar funções novas por omissão.

Papéis de acesso

Exercício resolvido • Aplique a matriz funcional

Enunciado e dados de partida

Defina as permissões para recepção, gestão e suporte. Recepção consulta agenda e registra check-in. Gestão consulta e edita agenda. Suporte consulta incidentes. Uma conta pode acumular papéis. A verificação deve recusar ações e papéis sem concessão explícita.

1. Salve politica.mjs

```
import assert from 'node:assert/strict';

const permissoes = new Map([
  ['recepcao', ['agenda:consultar', 'reserva:checkin']],
  ['gestao', ['agenda:consultar', 'agenda:editar']],
  ['suporte', ['incidente:consultar']],
]);

function permite(papeisAtivos, acao) {
  return papeisAtivos.some(papel =>
    permissoes.get(papel)?.includes(acao) === true
  );
}

const casos = [
  [['recepcao'], 'reserva:checkin', true],
  [['recepcao'], 'agenda:editar', false],
  [['gestao'], 'agenda:editar', true],
  [['gestao'], 'reserva:checkin', false],
  [['suporte'], 'incidente:consultar', true],
  [['suporte'], 'agenda:consultar', false],
  [['gestao', 'recepcao'], 'reserva:checkin', true],
  [[], 'agenda:consultar', false],
  [['papel-desconhecido'], 'agenda:consultar', false],
  [['gestao'], 'reserva:estornar', false],
];

for (const [papeis, acao, esperado] of casos) {
  assert.equal(permite(papeis, acao), esperado, acao);
}
console.log(casos.length + ' decisoes conferidas');
```

2. Execute

Rode node politica.mjs na pasta do arquivo. O Node.js já fornece assert. A saída esperada é ‘10 decisoes conferidas’.

Papéis de acesso

Revise concessões e revogações

Explique o resultado

O mapa guarda as ações de cada papel. some procura uma atribuição ativa que autorize a ação. No suporte, consultar incidente passa e consultar agenda falha. Uma conta com gestão e recepção pode fazer check-in pela segunda atribuição. O exercício presume que o código que chama a função já restringiu os papéis ao contexto aplicável.

Caso de erro • Concessão ampla por comodidade

Adicione 'agenda:editar' a recepcao e rode novamente. A asserção desta ação para recepção falha. Restaure o mapa e confira o sucesso. Para exigir dupla aprovação de estorno no futuro, especifique essa regra de negócio além dos papéis que podem participar.

Variação • Retire uma atribuição

Simule uma conta que começou com ['gestao', 'recepcao'] e perdeu recepcao. Na nova lista ['gestao'], a edição da agenda permanece permitida, e check-in passa a ser recusado. A mudança afeta a atribuição daquela conta. As permissões do papel continuam disponíveis a outros atendentes.

Critérios para conferir

Confira os resultados positivos, desconhecidos e sem papel. Teste a concessão indevida isoladamente e restaure a política original. Em uma integração, use uma identidade autenticada e confira papéis vigentes, unidade autorizada e a reserva concreta. A função de demonstração só decide a permissão funcional.

Referências primárias

[1] NIST • RBAC, papéis e atribuição de permissões

[2] OWASP • Menor privilégio e recusa por padrão

Papéis de acesso

Atribua permissões ao papel e ao escopo

Modele a atribuição no servidor

Associe a ação confirmar reserva ao papel de recepção e limite cada atribuição à unidade correspondente. A interface pode ocultar controles, mas a API avalia papel e unidade a cada pedido.

Exercício adicional

Uma pessoa é recepcionista na unidade A e apenas cliente na B. Onde pode confirmar reservas feitas por terceiros?

Resolução

Apenas na unidade A. Teste com uma reserva fictícia de cada unidade e confirme que o papel de recepção de A não concede a mesma permissão em B.

Ao assistir ao vídeo

Compare as decisões baseadas em papel com as que também dependem de atributos do recurso.

Vídeo complementar

IBM Technology - RBAC e ABAC (em inglês)