Engineer

Contrôle d'accès

accepter ou refuser chaque demande d'accès à une ressource

Ce que c’est

Le contrôle d’accès est la partie d’un système qui répond de nombreuses fois par seconde à une question : cet appelant peut-il effectuer cette action sur cette ressource, maintenant? L’appelant peut être une personne, un service ou une tâche planifiée. La ressource peut être un fichier, une ligne de base de données, un point d’entrée d’API, un dépôt git ou une porte. Les mécanismes individuels qui répondent sont appelés contrôles d’accès, et le glossaire du NIST donne l’autorisation comme synonyme du concept général. Ses définitions couvrent les deux aspects, traduits ici : « la décision d’autoriser ou de refuser à un sujet l’accès aux objets du système » et « le processus d’acceptation ou de refus de demandes précises » pour obtenir et utiliser des informations et les services de traitement associés, ou entrer dans certaines installations physiques.

On le confond facilement avec l’authentification, alors qu’il s’agit de deux étapes distinctes. NIST SP 800-162 le dit clairement : « L’authentification n’est pas la même chose que le contrôle d’accès ». L’authentification établit qui pose la question. Le contrôle d’accès prend cette identité, avec les autres informations connues sur la requête, et décide des actions permises. Un système peut vérifier parfaitement l’identité d’une personne tout en lui laissant lire les dossiers d’autrui, parce que la seconde question n’a jamais été posée.

Le même guide divise le travail en deux. Un point de décision de politique calcule la réponse, et un point d’application de politique se trouve sur le parcours de la requête pour l’exécuter. Les deux peuvent résider dans un seul processus, mais la distinction mérite d’être gardée à l’esprit : une décision que rien n’applique ne protège rien. NIST SP 800-53 en fait deux contrôles : AC-24 demande qu’une décision soit prise pour chaque demande d’accès avant son application, et AC-3 exige l’application des autorisations approuvées.

Les règles peuvent prendre plusieurs formes : une liste d'identités autorisées pour chaque objet, des rôles comme dans RBAC, ou des attributs de l'appelant, de l'objet, de l'action et de l'environnement, sujet de SP 800-162. Quelle que soit la forme, quelque chose doit encore poser la question à chaque requête.

Utilisez le concept pour prendre une décision

Utilisez « Qu'est-ce que le contrôle d'accès ? » pour éclairer une décision réelle. Nous pouvons vous aider à relier ce concept au système que vous construisez ou achetez. Des comptes, mots de passe et droits attribués selon les besoins exacts de chaque poste, puis retirés le jour du départ.

Découvrir nos prestations Contactez-nous →

Vérifiez les hypothèses derrière le terme

Appuyez-vous sur l'explication et les réalisations liées pour formuler les questions de votre projet. Nous pouvons examiner les hypothèses avec vous avant le choix d'un outil. 1 000 règles RBAC pour sécuriser Linux — 1,000 règles d'accès écrites pour combler une faille de sécurité

1 000 règles RBAC pour sécuriser Linux Contactez-nous →

Pourquoi c'est important

Une fois la personne connectée, le contrôle d'accès décide de presque toutes les ressources accessibles. Une authentification forte devant un contrôle d'accès faible ne fait qu'enregistrer quel compte a pris les données. Pour une entreprise qui détient les informations d'autrui, c'est ici que la promesse sur les personnes autorisées à voir les données est tenue ou rompue.

Le NIST décrit le mécanisme d'application idéal comme un moniteur de référence. Le contrôle AC-25 de SP 800-53 en exige un inviolable, toujours invoqué et assez petit pour être analysé et testé. Ces trois propriétés offrent un test pratique pour tout contrôle dans un produit réel. Si l'appelant peut modifier les entrées dont dépend le contrôle, la première échoue. Si une route atteint les données sans passer par lui, la seconde échoue. Si la logique est dispersée entre des centaines de gestionnaires au point que personne ne peut la lire de bout en bout, la troisième échoue.

L'OWASP Application Security Verification Standard transforme cela en exigences testables par une équipe. Son chapitre sur l'autorisation demande de limiter l'accès aux fonctions aux appelants explicitement autorisés, de limiter de la même façon l'accès à des données précises et d'appliquer les règles dans une couche de service de confiance, sans dépendre d'éléments manipulables par un client non fiable, comme du JavaScript dans le navigateur. Il demande aussi que tout changement des valeurs dont dépend une décision prenne effet immédiatement ou, si ce n'est pas possible, qu'un contrôle compensatoire déclenche une alerte lorsqu'une personne agit après avoir perdu ce droit.

La contrepartie est un coût sur le parcours de la requête. Toute décision nécessite des faits et chaque fait une recherche. Éviter cette recherche tend à déplacer le contrôle là où il coûte le moins, souvent à l'endroit contrôlé par l'appelant.

À quoi cela ressemble en pratique

Sur une plateforme maritime où une personne peut agir pour plusieurs organisations, la question de l'autorité a été tranchée par le lieu de décision : l'organisation active est conservée côté client, sans que cette information soit considérée comme fiable pour les décisions d'accès. Un OrganizationContext stocke le choix courant dans localStorage pour permettre un changement instantané, et un cookie côté serveur le reflète afin que les pages rendues sur le serveur résolvent le même contexte. L'autorisation est revérifiée sur le serveur à chaque requête. La page résume la limite en une phrase : le serveur est la seule autorité sur vos droits, et modifier localStorage ne change que votre propre interface, car le serveur continue de refuser.

Sur le même type de plateforme, l'encaissement et l'octroi des permissions ont été construits comme deux systèmes. Les cartes via Stripe et les achats intégrés Apple et Google convergent vers un schéma de paiement de 9 tables et 34 fonctions. Un schéma d'accès distinct de 11 tables et 34 fonctions répond à « ce compte peut-il faire cela ? » sans connaître la boutique à l'origine du paiement. Cette réponse conditionne l'annuaire des entreprises, la recherche et l'export de données, et elle est revérifiée sur le serveur à chaque requête, car un bouton caché est une courtoisie, pas un contrôle. Ajouter une boutique ne touche désormais que la partie paiement, et une question d'assistance sur les accès ne nécessite qu'une table au lieu de trois. La page en indique le coût : deux schémas et une recherche des droits sur des requêtes qui ne nécessiteraient sinon pas cette recherche.

Lorsque l'entreprise a déplacé son code source sur son propre matériel, la forge git a été configurée pour refuser par défaut. La page cite le contrôle d'accès parmi les responsabilités liées à l'exploitation d'une forge. Celle-ci a été choisie pour son absence d'interface Web d'administration : moins il y a d'interface, moins il y a à défendre. La politique part du refus : un appelant sans identité n'obtient rien, un appelant sans clé n'obtient rien, et chaque dépôt déclaré porte l'indicateur privé. La confidentialité repose ainsi sur la politique plutôt que sur la dissimulation. L'accessibilité réseau relève de la même politique. Le port 23231

répond uniquement sur l'interface de bouclage, ce qui fait d'un bastion l'unique accès extérieur, et le multiplexeur qui pourrait exposer git sur le port public reste désactivé dans le code : un point d'écoute inutilisé est une surface exposée.

Où les problèmes surviennent

Le contrôle réside dans l'interface. Un bouton est caché ou grisé, mais son point d'entrée répond à quiconque l'appelle directement. Le serveur doit prendre à nouveau la décision à chaque requête à partir de faits que l'appelant ne peut pas modifier.

La fonction est vérifiée, pas l'enregistrement. Une requête désigne une facture, un document ou un compte par son identifiant, le serveur confirme que l'appelant peut consulter des factures et renvoie celle demandée sans en vérifier le propriétaire. ASVS appelle cette faille insecure direct object reference (IDOR) ou broken object level authorization (BOLA). La personne chargée de la défense la reconnaît à un identifiant présent dans une requête et utilisé tel quel par le serveur. La correction consiste à résoudre chaque objet dans le périmètre propre à l'appelant et, dans un produit à plusieurs organisations clientes, à prendre l'organisation dans la session serveur, jamais dans la requête.

Tout nouvel élément reste ouvert jusqu'à sa fermeture. Une route, une table ou un dépôt commence avec la règle prévue en l'absence de réglage explicite. Si celle-ci autorise, la faille persiste jusqu'à ce qu'on la remarque. SP 800-53 exprime la règle inverse pour le réseau : refus sans autorisation explicite, autorisation par exception. Elle convient à chaque couche : ce qui n'a jamais été accordé est refusé.

Les autorisations ne font que s'accumuler. SP 800-162 note que les utilisateurs accumulent des privilèges lorsque leurs accès ne sont pas supprimés ou révoqués au fil du temps. Les personnes changent d'équipe, les projets se terminent et les prestataires partent : autant de moments où quelque chose devrait être retiré. Une révocation qui ne prend effet qu'à l'expiration d'une session, des heures plus tard, est une version atténuée de la même faille.

Personne ne sait expliquer pourquoi une requête a été autorisée. Lorsque les décisions sont dispersées et jamais journalisées, une enquête n'a rien à lire. Consignez les autorisations et les refus avec la règle qui les a produits, et protégez ce journal : SP 800-162 avertit que la succession des décisions peut révéler des attributs des personnes derrière les requêtes.

Pour aller plus loin

- Le contrôle d’accès dans le glossaire NIST CSRC — définitions issues de FIPS 201-3 et CNSSI 4009, avec l’autorisation comme synonyme.
- NIST SP 800-53 Rev. 5 — la famille du contrôle d’accès, notamment l’application AC-3, les décisions AC-24 et AC-25.
- NIST SP 800-162 — le contrôle d’accès par attributs, avec ses points de décision et d’application de politique.
- OWASP Application Security Verification Standard — exigences d’autorisation testables aux niveaux des fonctions, des données et des champs.

Glossaire

Choisissez selon vos contraintes réelles

Présentez le parcours de déploiement, les risques opérationnels et les personnes d'astreinte. Nous pouvons définir un changement avec des vérifications, une procédure de reprise et une responsabilité claire.

Consulter les tarifs Contactez-nous →

Rendez la terminologie concrète

Demandez comment « Qu'est-ce que le contrôle d'accès ? » s'applique à votre organisation. Un exemple concret tiré de votre système actuel suffit pour engager la discussion.

Découvrir nos prestations

welcome@engineer.company +45 50 10 61 84 Copenhague, Danemark

Emportez cette page avec vous

Fiche de contact

Créé avec soin et amour par D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Droit d'auteur © 2025 – aujourd'hui · Engineer ApS · Tous droits réservés.

Imprimé depuis https://engineer.company/fr/glossary/access-control/