Engineer

Utilisateur

personne ou processus agissant pour elle et admis au système

Ce que c’est

Un utilisateur est la personne ou l’entité pour laquelle un système agit. Dans les textes techniques courants, le mot remplit trois fonctions différentes, et l’essentiel de la confusion vient de leur mélange.

Le premier sens est la personne : quelqu’un qui ouvre un produit pour accomplir quelque chose. Les concepteurs, le personnel d’assistance et les responsables produit emploient généralement ce sens, qui n’a aucune forme fixe à l’intérieur du système. Le deuxième est le compte : l’enregistrement qu’un système conserve pour reconnaître quelqu’un, généralement un nom de connexion, un identifiant interne, un ensemble de permissions et un état tel qu’actif ou suspendu. L’essentiel du code applicatif emploie ce sens. Le troisième est l’utilisateur du système d’exploitation : sur les systèmes de type Unix, une entrée dans la base des utilisateurs avec un nom et un identifiant numérique. Les processus en cours d’exécution portent des identifiants d’utilisateur que le noyau utilise pour les contrôles de permissions.

Les trois se correspondent rarement un pour un. Une personne peut avoir des comptes auprès de nombreux services, et les recommandations du NIST sur l’identité numérique disent explicitement qu’une personne peut posséder plusieurs identités numériques, chacune unique seulement dans son propre service. Un compte peut être partagé par plusieurs personnes, ce qui est presque toujours une erreur. Et de très nombreux utilisateurs ne correspondent à aucune personne : le serveur web, la base de données, la tâche de sauvegarde et le script de déploiement s’exécutent chacun sous leur propre utilisateur du système d’exploitation et possèdent souvent aussi des comptes dans d’autres systèmes. C’est pourquoi la définition du NIST dit « une personne, ou un processus (du système) agissant pour le compte d’une personne ».

Sur un hôte Unix, la distinction entre nom et numéro est facile à voir :

```
$ id ana
uid=1001(ana) gid=1001(ana) groups=1001(ana)
```

Le noyau travaille avec le numéro. POSIX définit une base des utilisateurs qui contient pour chacun le nom d'utilisateur, l'identifiant numérique, un identifiant de groupe initial, un répertoire de travail initial et un programme initial à exécuter. La propriété des fichiers est stockée sous forme de numéro ; le nom n'est recherché que pour afficher quelque chose à une personne. Un processus porte aussi plusieurs identifiants d'utilisateur : un réel, qui indique qui l'a lancé, et un effectif, qui indique à quel utilisateur appartiennent les permissions utilisées à cet instant.

Les comptes applicatifs font généralement la même séparation. Le standard SCIM, utilisé pour provisionner les utilisateurs d'un système dans un autre, impose un `userName` visible par les personnes et servant à la connexion, unique dans le service et comparé sans tenir compte de la casse. Il impose aussi un `id` distinct, attribué par le service, stable et jamais réattribué. Un indicateur `active` distingue un compte suspendu d'un compte utilisable.

Utilisez le concept pour prendre une décision

Utilisez « Qu'est-ce qu'un utilisateur ? » pour éclairer une décision réelle. Nous pouvons vous aider à relier ce concept au système que vous construisez ou achetez. Portable, imprimante, connexion et messagerie : des problèmes réglés rapidement, en termes simples, sans mettre personne mal à l'aise.

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. Création d'un service d'assistance technique — 10,000 clients servis par un service de support créé de toutes pièces

Création d'un service d'assistance technique Contactez-nous →

Pourquoi c'est important

Chaque décision d'accès est une décision sur un utilisateur : une conception confuse de son identité produit donc une décision confuse. La question « Cette personne peut-elle voir cette facture ? » devient dans le code « Ce compte, dans ce contexte, possède-t-il une permission couvrant cet enregistrement ? ». Chaque étape de cette traduction est un endroit où la personne visée et le compte réel peuvent diverger.

La frontière entre personne et compte détermine aussi à qui attribuer les actes. Les journaux enregistrent des comptes. Lorsque chaque compte appartient à une seule personne, une ligne de journal dit qui a fait quelque chose. Lorsqu'un compte est partagé, le journal dit quel secret a été utilisé, mais pas qui l'a saisi, et retirer l'accès à une personne impose de le changer pour tout le monde.

Les utilisateurs du système d'exploitation sont depuis longtemps un outil du moindre privilège. Un service qui s'exécute sous son propre utilisateur non privilégié est limité par les permissions de cet utilisateur, pas seulement par la propriété des fichiers. Il peut encore modifier des ressources partagées accessibles en écriture, effectuer des requêtes réseau et utiliser les informations d'authentification à sa disposition. Des utilisateurs de service distincts réduisent les dégâts possibles d'une compromission ; ils n'assurent pas une isolation complète. Le même service exécuté sous le superutilisateur peut mettre toute la machine en danger. Donner à chaque programme son propre utilisateur est une petite modification de configuration qui réduit les permissions disponibles en cas de problème.

Enfin, personnes et comptes ont des durées de vie différentes. Les personnes changent de nom, d'adresse électronique et d'employeur ; les comptes sont suspendus, fusionnés et supprimés. Un système qui utilise un attribut modifiable, comme une adresse électronique, comme identité du compte finira par rattacher l'historique d'une personne à une autre.

À quoi cela ressemble en pratique

Sur une plateforme maritime, des professionnels gèrent des organisations, comme des entreprises et des académies, qui n'ont pas de compte de connexion propre. La page énonce directement la règle : la personne est le compte, et l'organisation est l'entité au nom de laquelle elle agit. Le travail a consisté à créer un changement de contexte permettant à un utilisateur d'agir au nom de toute organisation qu'il gère. Le stockage local du navigateur conserve l'organisation active, ce qui rend le changement instantané, et un cookie côté serveur en reflète la valeur pour que les pages rendues sur le serveur montrent le même contexte. Rien de cela ne décide de l'accès. L'autorisation est revérifiée côté serveur à chaque requête : une personne qui modifie la valeur stockée ne change donc que ce que montre son propre écran.

Sur l'hôte de l'entreprise, les versions sont déployées par un utilisateur du système d'exploitation presque dépourvu de droits. Lors de la mise en place de la plateforme de conteneurs, le déploiement a été séparé du provisionnement : root prépare une fois le chemin de déploiement. Ensuite, un utilisateur non privilégié redéploie en exécutant un script fixe au moyen d'une règle limitée qui autorise cette commande sans aucun argument. Une livraison ordinaire construit l'image et redémarre le service, sans que personne ouvre une session shell root pour cela.

Savoir qui sont les utilisateurs fait aussi l'objet d'un rapport. L'un des sept rôles de rapport en lecture seule produit, à partir de l'hôte en fonctionnement, un rapport d'accès en Markdown dans le dépôt, réunissant comptes, périmètre sudo, empreintes de clés, utilisateurs de la forge Git et rôles de base de données. Les sept rapports suivent une même règle : un chiffre n'apparaît que si l'exécution l'a mesuré, et une lacune reste vide plutôt que remplie par une estimation plausible qu'un lecteur croirait. Ensemble, les rapports ont trouvé de vrais problèmes, dont un port à l'écoute non déclaré et deux contrôles de sécurité inopérants.

Où les problèmes surviennent

Un utilisateur supprimé revient sous une autre identité. Un compte est supprimé, son identifiant est attribué de nouveau et le nouvel arrivant hérite de tout ce qui pointe encore vers l'ancienne valeur : lignes de table, place dans une liste, fichiers sur disque. Sur un hôte Unix, les fichiers d'un utilisateur supprimé conservent leur propriétaire numérique, et le prochain utilisateur créé avec ce numéro les possède. Il faut donner aux comptes des identifiants jamais réutilisés, comme SCIM l'exige pour son `id`, et décider du sort des données d'un utilisateur qui part avant de le supprimer.

Une adresse électronique sert de compte. Elle semble être une clé naturelle jusqu'à ce qu'une personne change d'adresse, perde l'accès à une ancienne ou qu'un employeur réattribue une adresse à une nouvelle recrue. L'historique d'une personne se retrouve alors rattaché à une autre, ou une personne se retrouve avec deux comptes. Il faut conserver un identifiant interne stable comme clé et traiter l'adresse comme un attribut modifiable.

Tout le monde se connecte au même compte. Un compte administrateur partagé est rapide à créer et impossible à auditer. Lorsqu'une personne quitte l'équipe, son accès ne prend fin que si le secret change pour tout le monde, et le

journal nomme un rôle plutôt qu'une personne. Un compte par personne, avec des pouvoirs communs accordés par des groupes, des rôles ou une règle sudo étroitement délimitée, permet de garder une trace fidèle des actes et de retirer correctement les accès au départ.

Le navigateur décide des droits de l'utilisateur. Un rôle conservé dans le stockage local, un cookie ou un champ de formulaire masqué n'est qu'une proposition du client, modifiable par n'importe qui. Si le serveur lui fait confiance, la modification devient une attaque. Le client peut mémoriser le contexte par commodité ; le serveur doit vérifier lui-même les permissions, à chaque requête.

Un service s'exécute sous root parce que c'était plus simple. Tout fonctionne, mais une faille exploitable dans ce service peut mettre tout l'hôte en danger. Chaque service doit recevoir son propre utilisateur non privilégié, et tout droit supplémentaire doit être limité à une seule commande nommément désignée, plutôt qu'accordé de façon générale.

Pour aller plus loin

- Glossaire NIST CSRC : user — les définitions utilisées dans les publications de sécurité du NIST, chacune accompagnée de son document d'origine.
- POSIX.1-2024 Base Definitions : Definitions — le sens formel de base des utilisateurs, identifiant et nom d'utilisateur sur les systèmes de type Unix.
- RFC 7643 : SCIM Core Schema — la ressource User pour transférer des comptes entre services, avec `userName` , `id` et `active` .
- NIST SP 800-63-4 : Digital Identity Guidelines — comment une personne peut posséder plusieurs identités numériques et les rôles de demandeur (applicant), d'inscrit (subscriber) et de personne à authentifier (claimant).
- POSIX.1-2024 : File Access Permissions — l'accès par les classes de permissions propriétaire, groupe et autres, les contrôles d'accès alternatifs et les privilèges appropriés ; la propriété seule ne limite pas l'accès.

Glossaire

Choisissez selon vos contraintes réelles

Décrivez l'appareil ou le service, les symptômes observés et les données à préserver. Nous pouvons déterminer l'étape de diagnostic utile la plus sûre.

Consulter les tarifs Contactez-nous →

Rendez la terminologie concrète

Demandez comment « Qu'est-ce qu'un utilisateur ? » 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/user/