Engineer

It-bruger

en person eller proces på personens vegne med systemadgang

Hvad det er

En it-bruger er den, eller det, et system handler på vegne af. I almindelig teknisk sprogbrug udfylder ordet bruger tre forskellige roller, og det meste af forvirringen om brugere kommer af, at de tre glider sammen til én.

Den første betydning er personen: en, der åbner et produkt og vil have noget gjort. Designere, supportfolk og produktansvarlige mener som regel denne, og den har ingen fast form inde i systemet. Den anden er kontoen: den post, systemet gemmer for at kunne genkende nogen igen, typisk et navn at logge ind med, et internt id, et sæt rettigheder og en status som aktiv eller suspenderet. Det meste programkode mener denne. Den tredje er brugeren i styresystemet: på Unix-lignende systemer en post i brugerdatabasen med et brugernavn og et numerisk bruger-id. Kørende processer bærer bruger-id-oplysninger, som kernen bruger til at kontrollere rettigheder.

De tre passer sjældent sammen én til én. En person kan have konti i mange tjenester, og NIST's retningslinjer for digital identitet siger direkte, at en person kan have flere digitale identiteter, som hver kun er unik inden for sin egen tjeneste. Én konto kan deles af flere personer, og det er næsten altid en fejl. Og rigtig mange brugere har slet ingen person bag sig: webserveren, databasen, backupjobbet og udrulningsscriptet kører hver som deres egen bruger i styresystemet og har ofte også konti i andre systemer. Derfor lyder NIST's definition på et individ eller en systemproces, der handler på vegne af et individ.

På en Unix-vært er skellet mellem navn og nummer let at se:

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

Kernen arbejder med nummeret. POSIX definerer en brugerdatabase, der for hver bruger rummer brugernavnet, det numeriske bruger-id, et første gruppe-id, en første arbejdsmappe og et første program, der skal startes. Ejerskab af filer gemmes som nummeret, og navnet slås kun op, når noget skal vises for et menneske. En proces bærer desuden mere end ét bruger-id: et reelt, der fortæller, hvem der startede den, og et effektivt, der fortæller, hvis rettigheder den bruger lige nu.

Konti i applikationer laver som regel samme opdeling. SCIM's brugerskema, som bruges til at overføre brugere fra ét system til et andet, kræver et `userName`, som folk ser og logger ind med, unikt i tjenesten og sammenlignet uden hensyn til store og små bogstaver, og et separat `id`, som tjenesten selv tildeler, som forbliver stabilt og aldrig gives videre til en anden. Et `active`-felt skelner mellem en suspenderet konto og en, der er i brug.

Brug begrebet i en beslutning

Brug »Hvad er en it-bruger?« i en konkret beslutning. Vi kan knytte begrebet til det system, I bygger eller køber. Den bærbare, printeren, login og postkassen klaret hurtigt og i et sprog, man forstår, uden at nogen føler sig dum.

Se vores ydelser Kontakt os →

Undersøg antagelserne bag ordet

Brug forklaringen og det tilknyttede arbejde til at formulere spørgsmål til jeres projekt. Vi kan undersøge antagelserne sammen, før et værktøj vælges. Har etableret en teknisk supportafdeling — 10,000 kunder betjent af en supportafdeling bygget fra bunden

Har etableret en teknisk supportafdeling Kontakt os →

Hvorfor det betyder noget

Enhver adgangsbeslutning er en beslutning om en bruger, så et uklart billede af, hvem brugeren er, bliver til en uklar beslutning. Spørgsmålet »må denne person se denne faktura?« besvares i koden som »har denne konto i denne kontekst en rettighed, der dækker denne post?«, og hvert trin i den oversættelse er et sted, hvor den tilsigtede person og den faktiske konto kan glide fra hinanden.

Skellet mellem person og konto er også der, hvor ansvaret bor. Logge registrerer konti. Når hver konto tilhører præcis én person, fortæller en loglinje, hvem der gjorde noget. Når en konto deles, fortæller loggen, hvilken hemmelighed der blev brugt, og intet om, hvem der tastede den, og at fjerne én persons adgang betyder, at den skal ændres for alle.

Brugere i styresystemet er et veletableret redskab til mindste privilegium. En tjeneste, der kører som sin egen bruger uden privilegier, er begrænset af den brugers rettigheder, ikke kun af filejerskab. Den kan stadig ændre fælles ressourcer med skriveadgang, sende netværksforespørgsler og bruge legitimationsoplysninger, den har adgang til. Separate tjenestebrugere reducerer skaden ved en kompromittering; de giver ikke fuldstændig isolation. Den samme tjeneste kørt som superbruger kan sætte hele maskinen i fare. At give hvert program sin egen bruger er en lille konfigurationsændring, som indsnævrer de tilgængelige rettigheder, når noget går galt.

Endelig lever personer og konti forskelligt længe. Folk skifter navn, e-mailadresse og arbejdsgiver; konti bliver suspenderet, slået sammen og slettet. Et system, der bruger en egenskab, som kan ændre sig, for eksempel en e-mailadresse, som kontoens identitet, kommer en dag til at hæfte én persons historik på en anden.

Sådan ser det ud i praksis

På en maritim platform administrerer fagfolk organisationer, for eksempel virksomheder og akademier, som ikke har deres egne logins. Siden formulerer reglen direkte: personen er kontoen, og organisationen er noget, personen handler på vegne af. Opgaven var et kontekstskifte, der lader en bruger handle som enhver organisation, vedkommende administrerer. Browserens localStorage holder styr på, hvilken organisation der er aktiv, så skiftet sker øjeblikkeligt, og en cookie på serversiden spejler valget, så serverrenderede sider viser den samme kontekst. Intet af det afgør adgang. Autorisationen kontrolleres igen på serveren ved hver request, så den, der ændrer den gemte værdi, kun ændrer, hvad vedkommendes egen skærm viser.

På virksomhedens egen vært sendes releases ud af en bruger i styresystemet med næsten ingen rettigheder. Da containerlaget blev sat op, blev udrulning adskilt fra provisionering: root sætter udrulningsstien op én gang, og derefter udruller en bruger uden privilegier igen ved at køre ét fast script gennem en afgrænset regel, der tillader netop den kommando og ingen argumenter. En almindelig release bygger imaget og genstarter tjenesten, og ingen åbner en root-shell for at gøre det.

At vide, hvem brugerne er, er i sig selv en rapport. En af syv skrivebeskyttede rapportroller gengiver en adgangsrapport fra den kørende vært som Markdown i repositoryet, med konti, sudo-omfang, nøglefingeraftryk, forge-brugere og databaseroller side om side. Reglen for alle syv rapporter er, at der ikke udskrives

noget tal, som kørslen ikke har målt. Tilsammen har rapporterne fundet reelle ting, blandt andet to døde sikkerhedskontroller og en lyttende port, som ingen havde erklæret.

Hvor det går galt

En slettet bruger vender tilbage som en anden. En konto fjernes, dens id bliver senere udstedt igen, og den nye arver alt, hvad der stadig peger på den gamle værdi: rækker i en tabel, en plads på en liste, filer på disken. På en Unix-vært beholder de filer, en fjernet bruger ejede, deres numeriske ejer, og den næste bruger, der oprettes med det nummer, ejer dem. Giv konti id'er, der aldrig genbruges, sådan som SCIM kræver af sit `id`, og beslut, hvad der skal ske med en fratrådt brugers data, før brugeren fjernes.

En e-mailadresse bruges som kontoen. Den ligner en naturlig nøgle, indtil nogen skifter adresse, mister adgangen til en gammel, eller en arbejdsgiver giver en adresse videre til en ny medarbejder. Så hæftes én persons historik på en anden, eller én person ender med to konti. Brug et stabilt internt id som nøgle, og behandl adressen som en egenskab, der kan ændre sig.

Alle logger ind på den samme konto. Et fælles administratorlogin er hurtigt at sætte op og umuligt at revidere. Når nogen stopper, ophører vedkommendes adgang kun, hvis hemmeligheden ændres for alle, og loggen nævner en rolle i stedet for en person. Én konto pr. person, med fælles rettigheder givet gennem grupper, roller eller en snævert afgrænset sudo-regel, holder både loggen og fratrædelserne ærlige.

Browseren afgør, hvad brugeren må. En rolle gemt i localStorage, en cookie eller et skjult formularfelt er et forslag fra klienten, og alle kan redigere det. Stoler serveren på det, er redigeringen et angreb. Klienten må gerne huske konteksten af bekvemmelighed; serveren skal selv kontrollere rettighederne ved hver request.

En tjeneste kører som root, fordi det var nemmere. Alt virker, men en fejl i tjenesten, der kan udnyttes, kan sætte hele værten i fare. Giv hver tjeneste sin egen bruger uden privilegier, og giv ekstra rettigheder snævert, til én navngiven kommando, i stedet for bredt.

Læs mere

- NIST CSRC-ordliste: user — definitionerne fra NIST's sikkerhedspublikationer, hver med det dokument, den stammer fra.
- POSIX.1-2024 Base Definitions: Definitions — den formelle betydning af brugerdatabase, bruger-id og brugernavn på Unix-lignende systemer.
- RFC 7643: SCIM Core Schema — User-ressourcen til at flytte konti mellem tjenester, med `userName`, `id` og `active`.
- NIST SP 800-63-4: Digital Identity Guidelines — hvordan én person kan have flere digitale identiteter, og rollerne applicant, subscriber og claimant, som dokumentet kalder dem.
- POSIX.1-2024: File Access Permissions — adgang gennem rettighedsklasserne ejer, gruppe og andre, alternative adgangskontroller og passende privilegier; ejerskab alene afgrænser ikke adgangen.

Ordliste

Vælg ud fra jeres faktiske vilkår

Beskriv enheden eller tjenesten, de observerede symptomer og de data, der skal bevares. Vi kan fastlægge det sikreste nyttige undersøgelsesskridt.

Se, hvad det koster Kontakt os →

Gør begreberne praktiske

Spørg, hvordan »Hvad er en it-bruger?« gælder i jeres organisation. Et konkret eksempel fra jeres nuværende system er nok til at begynde samtalen.

Se vores ydelser

welcome@engineer.company +45 50 10 61 84 København, Danmark

Tag denne side med dig

Visitkort

Lavet med kærlighed af D.

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

Copyright © 2025 – nu · Engineer ApS · Alle rettigheder forbeholdes.

Udskrevet fra https://engineer.company/da/glossary/user/