Opgave 9 - Bruger kontoer i Linux

Information

Formålet med følgende øvelse er at gøre dig i stand til at oprette, ændre og fjerne brugere i Linux-systemet.

I Linux er det kun brugere med superbrugerrettigheder, der kan foretage ændringer i forhold til brugerkonti på systemet. Da emnet 'Brugerrettigheder' endnu ikke er blevet introduceret, vil der først være en kort gennemgang af superbrugerrettigheder i Linux. Formålet er, at du forstår, hvordan kommandoen 'sudo' kan anvendes.

Husk at notere alle jeres besvarelser på jeres cheatsheet.

Baggrundsinformation: Forståelse af brugersystemet i Linux

I Linux fungerer brugersystemet som en mekanisme til at styre, hvilke personer og processer der kan tilgå og ændre systemet. Hver bruger får tildelt en unik konto, og systemet holder styr på, hvilke filer og ressourcer brugeren kan få adgang til. For at sikre, at brugerne kun har de nødvendige rettigheder, bruges principper som **least privilege** og **accountability**.

- **Root-kontoen** giver fuld adgang til systemet, men det er ikke altid godt at bruge denne konto direkte.
- **Sudo-kommandoen** giver brugere mulighed for at udføre administrative handlinger med root-rettigheder, men under kontrollerede betingelser.
- **Grupper** hjælper med at organisere brugere og give dem rettigheder på tværs af flere konti.

I Linux systemer, omtales en brugerkonti som et *login*.

Før du begynder øvelsen, er det vigtigt at forstå disse grundlæggende koncepter, da de vil hjælpe dig med at forstå, hvordan Linux håndterer brugere og deres rettigheder.

De mere uddybende forklaringer kan findes i forberedelsen til idag.

Baggrunden for sudo-kommandoen

Når man arbejder i Linux, anvendes kommandoen *sudo* meget for at opnå privilegerede rettigheder med sin konto. Derfor får I en forklaring på den, inden I skal i gang med at oprette brugerkontoer.

Brugersystemet i Linux (og andre operativsystemer) har til formål at segmentere enkelte brugeres rettigheder, således at hver bruger kun kan udføre arbejde på operativsystemet inden for sin bemyndigelsesramme (tænk: principle of least privilege). Derudover giver brugersystemet mulighed for at sikre sporbarhed i forhold til hvilke operationer den enkelte bruger har udført på operativsystemet (Accountability, repudiation, audit). Derfor bør der for hver fysisk bruger, der tilgår et Linux-system, eksistere en særskilt brugerkonto. En brugerkonto i Linux omtales ofte som et login.

Linux-brugersystemet har som minimum altid én bruger kaldet "root", som også kan benævnes som "superbrugerkontoen". Root-kontoen har adgang til alt i et Linux-system, hvilket betyder, at den kan læse, slette, oprette og redigere alle filer samt mapper. Root-kontoens login kan og bør dog deaktiveres.

Det er dog ikke hensigtsmæssigt med kun én superbrugerkonto. Med udgangspunkt i, at hver fysisk bruger altid skal tilgå et Linux-system med en særskilt konto, opstår der en konflikt, hvis mere end én bruger har behov for superbrugerrettigheder. Sporbarheden i forhold til den enkelte brugers handlinger i systemet vil ikke kunne opretholdes. Derudover udgør en superbrugerkonto det, man kalder et "single point of failure" i konteksten af det enkelte Linux-system. Det vil sige, at hvis en konto, som har rettigheder til alt, bliver kompromitteret, er hele systemet kompromitteret. Risikoen for at en konto bliver kompromitteret gennem f.eks. passwordlæk eller lignende stiger, desto flere der har adgang til passwordet.

For at løse udfordringen med superbrugerrettigheder til flere brugere i Linux, bruger man "sudo" (Super User Do), som er en kommando, der kan bruges af alle, der er medlem af en gruppe kaldet "sudoers". Modsat *Root*-kontoen, som altid har adgang til alt, kan medlemmer af *sudo*-gruppens rettigheder begrænses til f.eks. enkelte kommandoer, filer, directories o.l. som må anvendes som superbruger.

Dette understøtter *Principle of Least Privilege* (POLP) på superbrugerniveau. Altså, selvom man f.eks. er medlem af sudo-gruppen, kan man ikke nødvendigvis tilføje nye brugere eller slukke for logsystemet på den enkelte host. I Ubuntu kan dette redigeres i filen */etc/sudoers*, og du kan læse mere om det i forberedelsen til i dag *Mastering Ubuntu Server* i *Kapitel 2* afsnittet *Configuring administrator access with sudo*.

I Ubuntu er den bruger, man indledningsvist opretter (typisk under installationen), altid medlem af gruppen kaldet "sudoers", med adgang til alt.

For at eksekvere en kommando med superbrugerrettigheder, eksekveres kommandoen således:
`sudo <Kommando>`.

Kommandoer til oprettelse, ændring og fjernelse af brugere

De følgende Linux-kommandoer, som er linket til herunder, skal bruges til at udføre øvelsen:

- useradd - opret en ny brugerkonto

- userdel - slet en brugerkonto
- usermod - ændring af en eksisterende brugerkonto
- passwd - ændring af brugerens adgangskode (blandt andet)
- su - skift til en anden brugerkonto

Instruktioner

I de følgende øvelser skal du løse en række opgaver relateret til brugerkonti. Alle opgaver kan løses med en af de kommandoer, som er nævnt ovenfor. Det er naturligvis tilladt at bruge informationssøgning, f.eks. på Google.

Husk hvad du har lært i tidligere øvelser.

Opret en ny bruger

I denne opgave skal der oprettes en ny bruger ved navn `Mandalorian`.
Som udgangs punkt oprettes der ikke et home directory til brugeren. Du bør derfor kigge i dokumentationen for at se hvad flaget `-m` gør

Tildel en bruger et password

I denne opgave skal brugeren `Mandalorian` have tildelt et nyt password.

Skift bruger

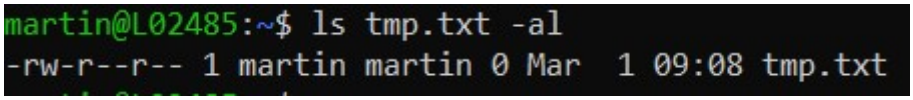
I denne opgave skal du skifte din aktive brugerkonto til `Mandalorian`.

Slet en bruger

I denne opgave skal brugeren `Mandalorian` slettes, men vi kigger samtidig på en finurlighed i Ubuntu vedrørende ejerskab af filer og sletning af bruger.

I Linux er alle filer ejet af en bestemt bruger. Typisk når en bruger opretter en fil, får denne bruger også tildelt ejerskab af filen.

Med kommandoen `ls <filenavn> -al` kan du få vist tilladelserne for en specifik fil. Som vist på billede nedenunder:



Formatet er som vist nedenunder:
| Ejerens rettigheder | - | Gruppens rettigheder | - | Alle andres rettigheder | Antal links til filen |

Ejer af filen | Tilknyttet gruppe | *Forklaring på resten af kolonnerne er bevidst undladt indtil videre*

1. Med brugeren `Mandalorian`, opret en fil (f.eks. en tom tekstfil).
2. Skift til en anden bruger og slet brugeren `Mandalorian` med `userdel`. (Dette kan drille, hvis der er en proces, som holder fast i brugeren. Kan du slukke processen?)
3. Find en af de filer, som blev oprettet af brugeren `Mandalorian`.
4. Se rettighederne for denne fil, og noter hvem der er filens ejer (Et navn?, et id?).
   Du kan ikke nødvendigvis skifte dig ind i Mandalorians directory, men er der en anden kommando du kan bruge til at se rettighederne for alle filer?
5. Opret en bruger ved navn `Ivan`.
6. Se igen rettighederne for den fil, der blev oprettet af `Mandalorian`. Hvem er ejeren nu?
7. Vurder om dette er en potentiel sårbarhed, og om man bør overveje at slette/skifte ejerskab på filer, når en bruger slettes.
   Et alternativ til at slette brugerkonti er at deaktivere dem.
8. Overvej om princippet *Secure by default* reelt er overholdt.

Links

- Et eksempel på, hvordan man kan lave krav til kodeords kompleksitet i Linux
- Grundlæggende brugerstyring i Linux
- useradd - opret en ny brugerkonto
- userdel - slet en brugerkonto
- usermod - ændring af en eksisterende brugerkonto
- passwd - ændring af brugers password (blandt andet)
- su - skift til en anden brugerkonto

2025-04-03 05:55:59