Øvelse 27 – Audit af en fil for ændringer

Information

Formålet med denne øvelse er at introducere, hvordan **audit-regler** kan konfigureres med **auditd**.
Auditd giver mulighed for at overvåge kritiske systemfiler og registrere ændringer, som kan være relevante for både sikkerhed og compliance.

I øvelsen arbejdes der med:

- Midlertidige audit-regler
- Permanente audit-regler
- Analyse af audit-logs

Baggrund

For at definere hvilke hændelser der skal logges af auditd, anvendes **audit-regler**.
Disse regler kan oprettes på to måder:

1. **Dynamisk** via kommandoen auditctl
2. **Permanent** via konfigurationsfiler i `/etc/audit/rules.d/`

Audit-logs gemmes som standard i følgende fil: `/var/log/audit/audit.log`

Auditd konfigurationsfiler

Auditd konfigureres primært gennem følgende filer og mapper:

- `/etc/audit/auditd.conf`

  > Indeholder generelle indstillinger for auditd, fx logfilens placering og størrelse.
- `/etc/audit/rules.d/`

  > Mappe med konfigurationsfiler for audit-regler.
  > Her kan man definere permanente regler, som indlæses ved opstart.
- `/etc/audit/audit.rules`

> Samlet fil med alle aktive regler.
> Denne fil genereres automatisk ud fra indholdet i `rules.d`.

 I praksis arbejder man typisk i `rules.d`, da det giver bedre struktur og overblik.

Eksempel på community- og standardregler

I praksis starter man sjældent med at skrive alle audit-regler selv.

Et eksempel på community-udviklede regelsæt kan findes i Linux Audit-projektets `rules/`-mappe på GitHub.
Her findes både basisregler og mere avancerede profiler, fx:

- `10-base-config.rules`
- `30-stig.rules`
- `31-privileged.rules`
- `40-local.rules`
- `99-finalize.rules`

Disse viser, hvordan auditd kan anvendes til fx:

- overvågning af systemændringer
- registrering af privilegerede handlinger
- compliance og hardening

Compliance og best practice (CIS)

Der findes også standarder og benchmarks for, hvordan auditd bør konfigureres.

Et vigtigt eksempel er **CIS Benchmark for Ubuntu Server**, hvor:

- **afsnit 4.1.3** indeholder konkrete anbefalinger til auditd-konfiguration

Disse anbefalinger bruges i praksis til at:

- sikre systemer mod kendte trusler
- opfylde compliance-krav
- etablere et baseline sikkerhedsniveau

 I dette forløb arbejder vi med simple regler for at forstå principperne, før vi ser på mere avancerede eller standardiserede regelsæt. I en senere øvelse skal der arbejdes med CIS

benchmarks

Instruktioner

Husk at notere observationer og resultater i dit Linux cheat sheet.

Brug af generative AI'er i undervisningen

Følgende tilgang må kun anvendes i undervisningssammenhæng (aldrig i en virksomhed).

Generative AI'er kan anvendes til at forklare output fra logs i en læringskontekst.
Hvis du er i tvivl om betydningen af et audit-log output, kan du stille følgende type spørgsmål:

> Jeg brugte denne kommando: `<kommando>`
> Jeg fik dette output: `<log output>`
> Kan du forklare, hvad dette betyder?

Vigtigt:
I en virksomhedsmæssig kontekst må interne logs eller følsomme data **aldrig** deles med generative AI-værktøjer, medmindre andet er anvist.

---

1 Tilføj auditregel med auditctl (midlertidig regel)

1. Opret en auditregel, der overvåger ændringer og skrivninger til filen /etc/passwd:
   `auditctl -w /etc/passwd -p wa -k user_change`
   Forklaring:
   - w angiver hvilken fil der overvåges
   - p definerer handlinger (w = write, a = attribute change)
   - k tildeler en nøgle, som bruges til søgning i logs
2. Udskriv en rapport over loggede hændelser: `aureport -i -k | grep user_change`
   - i oversætter numeriske værdier til læsbare navne
   - k filtrerer på nøgle

     > Der bør være en hændelse i loggen. Det er tilføjelsen af reglen.
3. Opret en ny bruger.
4. Udskriv rapporten igen og verificér, at der nu er to nye hændelser:
   - Én for skrivning til filen (w)
   - Én for ændring af metadata (a)

5. Analysér hændelserne mere detaljeret med: `ausearch -i -k user_change`

> Aureport giver hændelse rapportering, der viser en enkelt hændelse, udledt af aggregering af flere logs.

---

2 Permanente auditregler via konfigurationsfil

Auditctl-regler er ikke persistente og forsvinder ved genstart.
For at gøre regler permanente skal de gemmes i /etc/audit/rules.d/.

1. Åbn filen `/etc/audit/audit.rules og gennemse indholdet`.
   Bemærk at filen autogenereres ud fra /etc/audit/rules.d/.
2. Gem de nuværende aktive regler i en ny fil: `sh -c "auditctl -l > /etc/audit/rules.d/custom.rules"`
   Tip: Kontrollér inden, at reglen stadig er aktiv: `auditctl -l`
3. Genstart auditd: `systemctl restart auditd`
4. Udskriv `/etc/audit/audit.rules` og verificér, at reglen nu er indlæst.
5. Opret endnu en bruger og verificér, at ændringen logges.
6. Slet filen `/etc/audit/rules.d/custom.rules` for at fjerne den permanente regel og genstart auditd.

---

3 Overvågning af en almindelig tekstfil

Auditd kan også anvendes til overvågning af almindelige filer.

1. Opret en testfil: `echo "Dette er en testfil" > testfil.txt`
2. Opret en auditregel: `auditctl -w /path/to/testfil.txt -p wa -k textfile_watch`
3. Tilføj en ny linje til filen: `echo "Ny linje tilføjet" >> testfil.txt`
4. Verificér hændelsen: `ausearch -i -k textfile_watch`
5. Verificer at hændelsen ikke (eller kun i begrænset omfang) bliver vist med aureport

> Ændringen af filen bør kun være synlig igennem ausearch, og ikke aureport. aureport aggregerer audit-events til overskuelige rapporter (fx per executable eller hændelsestype) og viser derfor ikke nødvendigvis alle individuelle filevents. ausearch kan sammenlignes med et direkte kig i `/var/log/audit/audit.log`. ofte omtales dette som forskel i granularitet

 Bemærk: Når en tekstfil ændres via shell (fx echo >> fil), vil ændringen typisk blive udført af shellen (bash). Dette gør, at hændelsen ikke fremstår som en tydelig applikationshændelse i aureport, men stadig logges korrekt i audit-loggen.

 Faglig pointe: Forskellen mellem aureport og ausearch illustrerer forskellen mellem aggregeret loganalyse og rå loganalyse, hvilket er centralt i både sikkerhedsovervågning og fejlsøgning.

Refleksion

Overvej følgende spørgsmål:

- Hvordan kan auditd hjælpe med at opdage uautoriserede ændringer?
- Hvilke filer eller systemområder bør overvåges på en produktionsserver?
- Hvordan kan auditd integreres med et SIEM-system som Wazuh?

Links

- Auditctl man page
- Linux Audit project

Last update: 2026-03-20 13:58:28