Engineer

Backup

en kopi af data mod tab, og det at lave den

Hvad det er

En backup er det, data gendannes fra, når originalen er slettet, ødelagt eller tabt sammen med den maskine, den lå på, så den skal ligge et sted, de hændelser ikke når. NISTs ordliste definerer den som en kopi af filer og programmer, lavet for at lette gendannelse. På dansk hedder den også en sikkerhedskopi, og handlingen hedder sikkerhedskopiering eller at tage backup: et natligt job tager backup af databasen og efterlader en backup.

Nabobegreber bliver forvekslet med den. En replika modtager hver ændring inden for sekunder, hvilket beskytter mod en død disk og kopierer en fejlagtig `DELETE` lige så trofast. Et snapshot fastfryser et volumen i ét øjeblik og bliver først en backup, når det ligger et sted, originalens fejl ikke kan følge med til. En backup defineres af historik og adskillelse: den giver data tilbage, som de så ud på et tidligere tidspunkt, fra et sted, der overlevede det, der tog originalen.

Databaser gør det konkret. PostgreSQL-manualen beskriver tre grundlæggende forskellige tilgange. Et SQL-dump, lavet med `pg_dump`, skriver de sætninger ud, der genopbygger databasen, som et konsistent øjebliksbillede af tidspunktet, hvor dumpet begyndte. En backup på filsystemniveau kopierer selve datamappen. Kontinuerlig arkivering kombinerer sådan en basebackup med den write-ahead log (WAL), serveren producerer bagefter, og når loggen afspilles, kan databasen gendannes til et hvilket som helst tidspunkt siden basebackuppen. Den evne hedder point-in-time recovery.

Brug begrebet i en beslutning

Brug »Hvad er en backup?« i en konkret beslutning. Vi kan knytte begrebet til det system, I bygger eller køber. Sikkerhedskopier, der er testet og ikke bare antaget at virke, og en nedskrevet plan for den morgen, en server ikke starter igen.

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 automatiseret 100 kritiske databackups — 100 kritiske backups af databaserne automatiseret

Har automatiseret 100 kritiske databackups Kontakt os →

Hvorfor det betyder noget

To tal gør et generelt ønske om sikkerhed til et design. NIST definerer recovery point objective, RPO, som det tidspunkt, data skal kunne gendannes til efter et udfald: i praksis, hvor meget af det seneste arbejde virksomheden har råd til at miste. Recovery time objective, RTO, er, hvor længe et system kan være under gendannelse, før udfaldet skader det, organisationen er der for at gøre.

De to trækker i hver sin del af designet. RPO bestemmes af, hvor ofte der tages en kopi: et natligt dump kan miste en dags skrivninger, mens WAL-arkivering skrumper tabet til de sidste øjeblikke af log, der stadig er på vej til arkivet. RTO bestemmes af, hvor lang tid en gendannelse tager. PostgreSQL-manualen bemærker, at serveren afspiller hvert WAL-segment siden den seneste basebackup, hvilket kan tage et stykke tid, når den backup er gammel. Hvor ofte der tages basebackups, er derfor lige så meget en RTO-beslutning som et spørgsmål om lagerplads.

Begge tal er påstande om en gendannelse, og derfor bedømmes en backup ved at gendanne den. Et job, der afslutter med status nul, viser, at der er skrevet bytes et sted. Om de genopbygger et fungerende system inden for RTO'en, afgøres ved at gendanne efter en fast plan, et andet sted end produktion, og sammenligne det, der kommer tilbage.

Adskillelse er den anden halvdel. En kopi i samme konto, under samme legitimationsoplysninger, deler de fejl, den skal overleve: en lukket konto, en stjålet nøgle, et script kørt mod det forkerte mål. Det nyttige spørgsmål til enhver kopi er, hvilken enkelt hændelse der ville tage den med sig sammen med originalen.

Sådan ser det ud i praksis

I den lille ende blev virksomhedens implementering af sikkerhedskopiering og gendannelsestest udformet til at give gendannelsen et grundlag ud over udbyderens snapshot. En Ansible-rolle installerer restic-scripts og systemd-units til et databasedump, hvor det er konfigureret, krypterede og deduplikerede snapshots, beskæring efter opbevaringsregler og integritetstjek. Et særskilt

månedligt forløb gendanner en kendt fil til et midlertidigt sted og sammenligner bytes. Sikkerhedskopiering og gendannelse har hvert sit successignal, og timerne kræver et konfigureret lager. Adgangssætningen læses fra en beskyttet fil frem for en miljøværdi, som systemd fortolker. Det er implementerede kontroller og forudsætninger for aktivering: implementeringsregistreringen dateret 2026-10-27 siger, at timerne var deaktiveret uden et lager, og registrerer ingen gennemført gendannelse under reelle driftsforhold. Det underbygger automatiseringsarbejdet, ikke en påstand om, at gendannelse er lykkedes, eller at værten allerede kan mistes uden at bringe data i fare.

I større skala blev hundrede kritiske databackups automatiseret med Barman og Google Cloud til et produkt, der lever af sine GIS-data. Barman tog sig af PostgreSQL, Google Cloud holdt offsite-kopierne, og Bash og Python orkestrerede og validerede kørslerne. Integritetstjek bekræftede, at hver backup var i orden og ikke bare til stede.

På endnu et produkt var opgaven at definere den gendannelige position i stedet for at gå ud fra den. Backup-retention ligger i Bicep-skabelonen ved siden af databasen, så point-in-time restore er en egenskab ved skabelonen frem for en indstilling, nogen engang klikkede i en portal, og miljøet omkring den er også kode. Auditten skrev hullerne ned: geo-redundans er slået fra, og RTO'en er foreslået frem for målt, med den øvelse, der skal måle den, beskrevet og klar til at køre.

Hvor det går galt

Problemer med backups har det med at forblive skjulte, til der skal gendannes, så hvert af dem er værd at lede efter en stille dag.

En filkopi af en kørende database gendannes som en ødelagt klynge. Kopierer man PostgreSQLs datamappe med `tar`, mens serveren kører, samler man filer fra forskellige øjeblikke, og manualen siger, at serveren skal lukkes ned for at få en brugbar backup på den måde. Et filsystem-snapshot taget i ét atomart øjeblik virker, hvis det indeholder WAL'en, som serveren afspiller ved opstart ligesom efter et nedbrud; ligger data og WAL på forskellige filsystemer, kan et samtidigt snapshot være umuligt. `pg_dump`, eller en basebackup til kontinuerlig arkivering, giver en konsistent kopi, mens serveren kører videre.

Et hul i WAL-arkivet afslutter point-in-time recovery ved hullet. Gendannelse kræver en ubrudt række af arkiverede WAL-filer, der rækker tilbage til starten af basebackuppen. Bliver arkivkommandoen ved med at fejle, prøver PostgreSQL igen og igen, og WAL hober sig op i `pg_wal/`, så det første symptom ofte er en disk, der fyldes. Manualens råd er at sætte arkiveringen op og teste den før den første basebackup og at overvåge den for forsinkelser.

Jobbet stopper, og stilheden ligner succes. Et timerjob, der går i stykker, skriver sin fejl til en log, som ingen læser, og hver stille dag går for en god dag. Overvågningen skal behandle stilhed som alarmen: et check-in, der kun sendes efter en vellykket kørsel, til en tjeneste, der giver besked, når det er forsinket, fanger et job, der er holdt op, lige så vel som et, der fejler højlydt.

Nøglen forsvinder, eller den ankommer ændret. Kryptering gør adgangen afhængig af nøglerne. restic understøtter flere adgangskoder til ét sikkerhedskopilager; advarslen om uigenkaldeligt datatab gælder, når der ikke er nogen brugbar nøgle til lageret tilbage. Nøglen skal ligge et sted, der overlever maskinen. I restic-opsætningen ovenfor læses adgangssætningen fra en fil i stedet for fra miljøet, fordi supervisoren behandler escape-tegn i miljøværdier, og en adgangssætning med en omvendt skråstreg ville være nået frem som andre bytes end dem, backuplageret blev oprettet med, uden nogen advarsel.

Opbevaringspolitikken sletter den kopi, der skulle bruges. Korruption opdages ofte uger efter, den skete, og en politik, der beholder syv daglige kopier, har til den tid erstattet hver ren kopi med en beskadiget. En trindelt politik beholder en lang hale til en lav pris, og restics `--dry-run` viser, hvad en politik ville fjerne, før noget forsvinder:

```
restic forget --keep-daily 7 --keep-weekly 5 --keep-monthly 12 \
  --prune --dry-run
```

`forget` fjerner snapshots, og `prune` fjerner de data, som kun de refererede til, så efter en prune er de gamle kopier væk for altid.

Integritetstjekket består, og gendannelsen går alligevel i stykker. Som standard kontrollerer `restic check` repositoryets struktur uden at læse hver pakkefil, fordi det ville betyde at downloade dem alle; `--read-data` læser det hele, og `--read-data-subset=1/5` læser en femtedel, så fem kørsler fra `1/5` til

`5/5` dækker det hele. En fuld læsning beviser kun, at de lagrede bytes er intakte. Om de genopbygger en fungerende database med de rigtige roller og indstillinger, viser en gendannelse.

Læs mere

- NIST glossary: backup — definitionen, der bruges her, og publikationerne bag den.
- NIST SP 800-34 Rev. 1 — vejledningen i beredskabsplanlægning, der definerer RPO og RTO.
- PostgreSQL: Backup and Restore — de tre tilgange, og hvad hver af dem koster.
- PostgreSQL: Continuous Archiving and Point-in-Time Recovery — basebackups, WAL-arkivering og gendannelse til et valgt tidspunkt.
- restic: Preparing a new repository — adgangskoder, adgangskodefiler og advarslen om at miste dem.
- restic: Removing backup snapshots — opbevaringspolitikker, `forget`, `prune` og `--dry-run`.

Ordliste

Vælg ud fra jeres faktiske vilkår

Medbring udrulningsforløbet, driftsrisiciene og dem, der har vagten. Vi kan afgrænse en ændring med kontroller, en vej tilbage og tydeligt ansvar.

Se, hvad det koster Kontakt os →

Gør begreberne praktiske

Spørg, hvordan »Hvad er en backup?« 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/backup/