Engineer

virtuell maskin

en hel datamaskin som kjører i programvare på en fysisk vert

Hva det er

En virtuell maskin presenterer en hel datamaskin gjennom programvare: prosessorer, minne, disker og nettverkskort levert av et virtualiseringslag på en fysisk vert, mens et operativsystem starter på dem som om de var dedikert maskinvare. Den forkortes ofte til VM, etter det engelske virtual machine. Noen enheter kan overføres direkte fra verten i stedet for å bli emulert. NISTs veiledning om full virtualisering, SP 800-125, definerer den på én linje som et simulert miljø skapt ved virtualisering. I den formen for virtualisering veiledningen handler om, kjører ett eller flere operativsystemer med programmene sine på virtuell maskinvare, hvert i sin egen VM, der det kalles et gjesteoperativsystem.

To ord gjør mesteparten av jobben rundt den. Den ekte maskinen er verten, og hver virtuell maskin på den er en gjest. Programmet som deler verten mellom gjestene, er hypervisoren, også kalt virtual machine monitor, som SP 800-125 beskriver som komponenten som styrer gjesteoperativsystemene på en vert og kontrollerer flyten av instruksjoner mellom dem og den fysiske maskinvaren: prosessor, disker, minne og nettverkskort. Samme veiledning peker på to ting som gjør ordningen nyttig. Hypervisoren kan dele opp vertens ressurser slik at hver gjest får tilgang til sine tildelte ressurser og eventuelle ressurser som er uttrykkelig delt, og en gjest kan pakkes helt inn, slik at den kan flyttes fra én vert til en annen.

En virtuell maskin skiller seg fra en container i hvor grensen går. En container er en prosess på vertens kjerne med et begrenset syn på systemet. En virtuell maskin starter en egen kjerne på maskinvare som hypervisoren gir den. Derfor kan en Linux-gjest kjøre på en Windows- eller macOS-vert, og derfor trenger hver gjest sitt eget operativsystem, som må installeres, oppdateres og få nok minne til å kjøre.

Bruk begrepet i en beslutning

Bruk «Hva er en virtuell maskin?» i en konkret beslutning. Vi kan knytte begrepet til systemet dere bygger eller kjøper. Servere dimensjonert etter det du faktisk bruker, flyttet inn i skyen eller ut av den igjen, med månedsregningen regnet ut på forhånd.

Se tjenestene våre Ta kontakt →

Undersøk antakelsene bak ordet

Bruk forklaringen og det tilknyttede arbeidet til å formulere spørsmål for prosjektet. Vi kan undersøke antakelsene sammen før et verktøy velges. Driftet 20 Docker-apper på Google Cloud og AWS — 20 containeriserte applikasjoner produksjonssatt og overvåket

Driftet 20 Docker-apper på Google Cloud og AWS Ta kontakt →

Hvorfor det er viktig

En server leid hos en skyleverandør er som regel en virtuell maskin, og infrastruktur som tjeneste er bygd på dem. Leverandøren driver hypervisorene og den fysiske maskinparken; kunden får maskiner som kan opprettes, endres i størrelse og slettes gjennom et API i stedet for å bestilles, leveres og monteres i et rack. Maskinvaren forblir andres problem, mens operativsystemet blir kundens.

Isolasjon er den andre grunnen. KVM-prosjektets side sier at hver virtuell maskin har privat virtualisert maskinvare, med eget nettverkskort, egen disk og eget grafikkort. Gjestene deler ingen kjerne, så et kjernekrasj eller en mislykket oppdatering i én gjest blir værende der, og det er hypervisoren, ikke gjestene, som bestemmer hva som når den fysiske maskinvaren. To kunder, eller to miljøer hos samme kunde, kan dele en fysisk maskin uten å dele operativsystem.

For det tredje kan en maskin med imagebasert lagring bli til et sett med filer. Disken til en gjest kan være en imagefil på verten, som kan kopieres, tas vare på som mal eller brukes til å starte flere like maskiner. QEMUs qcow2-format kan registrere bare forskjellene fra et basisimage og kan lagre øyeblikksbilder av disken i den samme filen, så én mal pluss et tynt lag per gjest er et naturlig oppsett.

For det fjerde lar emulering programmer som er bygd for én prosessorfamilie, kjøre på en annen. Systememuleringen i QEMU gir en virtuell modell av en maskin, med prosessor, minne og enheter, som et gjesteoperativsystem kan kjøre på. Med en akselerator, som KVM på Linux, Hypervisor Framework på macOS eller Windows Hypervisor Platform på Windows, tar virtualiseringsstøtten i vertens prosessor det tunge arbeidet. Uten en akselerator oversetter QEMUs

Tiny Code Generator gjestens instruksjoner i programvare, og det er slik introduksjonen til QEMU-dokumentasjonen starter en Arm64-maskin i sitt eget eksempel, med `-accel tcg` .

Prisen er at hver gjest er en fullverdig datamaskin som må passes på. Hver har sitt eget operativsystem som skal oppdateres, sitt eget reserverte minne og sin egen oppstart. Containere unngår det ved å dele vertens kjerne, noe som gjør hver av dem lettere og gir opp en egen kjerne som veggen mellom arbeidslastene. Å velge mellom dem er å velge hvor den veggen skal stå.

Slik ser det ut i praksis

På x86-Linux er det vanlige paret KVM og QEMU. KVM er kjernedelen: en lastbar modul, `kvm.ko` , med en prosessorspesifikk følgesvenn, `kvm-intel.ko` eller `kvm-amd.ko` , og den har vært med i hovedlinjen av Linux siden 2.6.20. Den trenger prosessorens virtualiseringsutvidelser, Intel VT eller AMD-V. KVMs FAQ ser etter flaggene deres, `vmx` og `svm` , i `/proc/cpuinfo` , og feilsøkingslisten der omfatter å bekrefte at `/dev/kvm` finnes, og at brukeren har rettigheter til å åpne den:

```
grep -E -c '^flags.*(vmx|svm)' /proc/cpuinfo
ls -l /dev/kvm
```

Et tall over null betyr at prosessoren melder om utvidelsene. QEMU er delen i brukerrommet, som bygger maskinen rundt KVM. Disken til en ny gjest kan være et tynt lag oppå et delt basisimage:

```
qemu-img create -f qcow2 -b debian-base.qcow2 -F qcow2 web1.qcow2
qemu-img info --backing-chain web1.qcow2
```

Håndboken for qemu-img sier at et image som er opprettet oppå et basisimage, bare registrerer forskjellene fra det, og at basisimaget aldri endres med mindre forskjellene føres tilbake i det med `qemu-img commit` . Den andre kommandoen viser hver fil i kjeden, og utdataene bør leses før noe flyttes eller slettes. Selve oppstarten av gjesten ser slik ut:

```
qemu-system-x86_64 -accel kvm -m 2G -smp 2 \
    -drive file=web1.qcow2,if=virtio,format=qcow2 \
    -nographic
```

`-accel kvm` gir prosessorarbeidet til KVM, som standard gjennom `/dev/kvm`. `-m` setter gjestens minne og `-smp` antallet prosessorer. På `-drive` kobler `if=virtio` disken til gjennom virtio-grensesnittet, et av grensesnittene som håndboken lister opp, og `format=qcow2` oppgir formatet i stedet for å la QEMU finne det ut selv. `-nographic` slår av det grafiske vinduet, slik at QEMU blir et vanlig kommandolinjeprogram med gjestens serielle konsoll i terminalen, delt med QEMU-monitoren; Ctrl-a h viser hvordan man bytter mellom de to.

Med gjesten slått av tar det samme verktøyet øyeblikksbilder av disken og gjenoppretter dem. `-c` oppretter et, `-l` lister dem, og `-a` setter disken tilbake til en lagret tilstand:

```
qemu-img snapshot -c before-upgrade web1.qcow2
qemu-img snapshot -l web1.qcow2
qemu-img snapshot -a before-upgrade web1.qcow2
```

Dimensjonering er en designbeslutning, ikke et felt i et skjema. Porteføljeoppføringen om å drive selskapets offentlig tilgjengelige tjenester på én liten server dokumenterer et oppsett fra september 2026 med én prosessorkjerne, en disk på 10 GB og 512 MB minne, der programmene kan bruke omtrent 464 MB. Verten betjener sju domener over TLS og kjører også git-forgen, mens automatisering av sikkerhetskopiering er implementert separat; minnet er grensen som alle designvalg der må forholde seg til. Om den også kunne ta en onion-tjeneste på Tor, ble avgjort med et dokumentert døgn med stikkprøvemålinger, der den laveste avlesningen av tilgjengelig minne var rundt 310 MB. Innstillingen `-m` for hver gjest stiller det samme spørsmålet, og en måling besvarer det bedre enn et større abonnement.

Oppføringen om administrasjon av tretti Ubuntu Linux-VPS-instanser viser den andre enden av skalaen. Oppsettet av disse instansene hadde hopet seg opp gjennom årene uten noen plan, og konfigurasjonene hadde glidd fra hverandre. Siden ble de drevet som én samlet maskinpark: én herdet nettverkskonfigurasjon på hver instans og gjenoppretting som ble prøvd i praksis, ikke tatt for gitt.

Hvor det går galt

De fleste problemer med virtuelle maskiner kommer av å glemme at hver av dem er en ekte datamaskin som må passes på, og at en imagebasert disk også er en vanlig fil på en annen maskin.

KVM mangler når gjesten starter. I x86-oppsettet over hindrer manglende eller fastvaredeaktiverte prosessorutvidelser KVM-akselerasjon, og `/dev/kvm` kan mangle, så en gjest som startes med `-accel kvm`, stopper med en gang. KVMs FAQ advarer om at noen produsenter slår av VT i BIOS på en måte som ikke kan slås på igjen. Løsningen er å kjøre flaggsjekken over, se etter et valg i fastvaren som slår på virtualisering, og bekrefte at `/dev/kvm` finnes og tilhører en gruppe brukeren er med i; FAQ-en foreslår en gruppe som heter `kvm`.

En gjest som startes uten minnestørrelse, får 128 MiB. Det er standardverdien til `-m` i QEMU, og den gjelder uansett hvor mye verten har ledig, så en gjest kan starte og deretter svikte på måter som ligner feil i programvaren inne i den. Løsningen er å sette `-m` og `-smp` hver gang og dimensjonere dem etter målinger, slik oppføringen om den lille serveren avgjorde spørsmålet om onion-tjenesten med et døgns målinger.

Et image som endres mens gjesten kjører, kan bli ødelagt. Håndboken for qemu-img er klar: aldri bruk verktøyet til å endre et image som en kjørende virtuell maskin eller en annen prosess bruker, fordi det kan ødelegge imaget. Å ta et øyeblikksbilde av disken til en kjørende gjest med `qemu-img snapshot -c` er nettopp det. Løsningen er å slå av gjesten før qemu-img rører disken, og å gjøre det til et eget steg i hver driftsrutine der det skjer.

Øyeblikksbilder brukes i stedet for sikkerhetskopier. Øyeblikksbilder tatt med qemu-img ligger i den samme imagefilen, noe `qemu-img info` viser ved å liste dem sammen med disken. Når filen går tapt eller blir skadet, forsvinner alle øyeblikksbildene med den. Løsningen er å kopiere sikkerhetskopier til lagring på en annen maskin og teste gjenopprettingen, fordi en gjenoppretting som aldri er kjørt, ikke beviser noe.

Et basisimage endres eller flyttes under imagene som bygger på det. Et tynt image inneholder bare forskjeller, så det betyr ingenting uten nøyaktig det basisimaget det ble laget mot. Håndboken advarer om at hvis basisimaget byttes i usikker modus (unsafe) og den nye filen er feil fil, blir innholdet gjesten ser, ødelagt, og en relativ sti til basisimaget tolkes fra katalogen til det avledede

imaget selv, så å flytte den ene filen uten den andre bryter kjeden. Løsningen er å behandle basisimager som skrivebeskyttede og gi hver ny versjon et nytt navn, å lese `qemu-img info --backing-chain` før noe flyttes, og å kjøre `qemu-img commit` bare med vilje, siden kommandoen skriver inn i basisimaget og tømmer det avledede imaget.

Diskformatet gjettes ut fra selve imaget. Uten `format=` finner QEMU formatet ut fra innholdet i filen, og håndboken anbefaler `format=raw` for å unngå å tolke et formathode som ikke er til å stole på. Et image lastet ned fra et annet sted, eller et som en gjest har skrevet til, er nettopp et slikt image. Løsningen er å oppgi formatet på hver `-drive` -linje.

Hver gjest glir over i å bli et kjæledyr. Maskiner som settes opp for hånd, glir fra hverandre én endring av gangen til ingen to er like, og det var der VPS-flåten over sto før den kom under forvaltning. Løsningen er å bygge hver gjest fra det samme imaget og den samme konfigurasjonen, ha konfigurasjonen i versjonskontroll og forvalte maskinene som én flåte med én grunnlinje.

Les mer

- Guide to Security for Full Virtualization Technologies (NIST SP 800-125) — definisjonen av en virtuell maskin, hypervisorens oppgave og hvordan gjester holdes adskilt.
- KVM — prosjektsiden: moduler, prosessorutvidelser og første kjerneversjon.
- KVM FAQ — sjekken av vmx og svm, rettigheter til `/dev/kvm` og BIOS-innstillingen.
- QEMU System Emulation introduction — akseleratorene, Tiny Code Generator og en komplett eksempelkommando.
- QEMU invocation — alle valgene som er brukt over: `-accel` , `-m` , `-smp` , `-drive` , `-nographic` .
- QEMU disk image utility — qcow2, basisimager, øyeblikksbilder og advarselen om image i bruk.

Ordliste

Velg ut fra deres faktiske rammer

Ta med utrullingsløpet, driftsrisikoene og dem som har vakt. Vi kan avgrense en endring med kontroller, en vei tilbake og tydelig ansvar.

Se hva det koster Ta kontakt →

Gjør begrepene praktiske

Spør hvordan «Hva er en virtuell maskin?» gjelder i organisasjonen deres. Et konkret eksempel fra dagens system er nok til å starte samtalen.

Se tjenestene våre

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

Ta med deg denne siden

Visittkort

Laget med kjærlighet av D.

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

Opphavsrett © 2025 – nå · Engineer ApS · Alle rettigheter forbeholdt.

Skrevet ut fra https://engineer.company/nb/glossary/virtual-machine/