Teknologi Virtualisasi Studi kasus: QEMU (Machine emulator and virtualizer)

Henry Saptono, M.Kom
STT Terpadu Nurul Fikri
November 2020

What is Qemu ?

- Qemu (Quick Emulator)

  “QEMU is a generic and open source machine emulator and virtualizer. “

  QEMU is a FAST! processor emulator using dynamic translation to achieve good emulation speed.
- QEMU dibuat oleh Fabrice Bellard
- Qemu adalah free software dan didistribusikan dengan lisensi GPL.

What is Qemu ?

- QEMU adalah **emulator mesin** generik dan open source dan **virtualizer**.
- Saat digunakan sebagai **emulator**, QEMU mampu menjalankan sistem operasi / program yang dibuat untuk satu jenis mesin pada jenis mesin yang berbeda. Bagaimana ini dilakukan ? Yaitu menggunakan metode terjemahan biner.
- Dalam mode emulator, QEMU mengemulasi CPU melalui teknik **terjemahan biner dinamis** dan menyediakan satu set model perangkat. Jadi, ini diaktifkan untuk **menjalankan sistem operasi tamu** yang tidak dimodifikasi dengan arsitektur berbeda. Terjemahan biner diperlukan di sini karena kode tamu harus dijalankan di CPU host.
- Penerjemah biner yang melakukan pekerjaan ini dikenal sebagai **Tiny Code Generator (TCG)**; ini adalah kompiler **Just-In-Time**. Ini mengubah kode biner yang ditulis untuk prosesor tertentu ke yang lain (misalnya: ARM ke X86):

What is Qemu ?

- Bila digunakan sebagai **virtualizer**, QEMU mengeksekusi kode tamu secara langsung pada CPU host, sehingga mencapai kinerja asli. Misalnya, saat bekerja di bawah hypervisor Xen / KVM.
- Jika KVM adalah hypervisor yang mendasarinya, QEMU dapat memvirtualisasikan tamu yang terpasang seperti Power PC, S390, x86, dan seterusnya.
- QEMU mampu berjalan tanpa KVM, menggunakan metode terjemahan biner yang disebutkan sebelumnya. Eksekusi ini akan lebih lambat jika dibandingkan dengan virtualisasi akselerasi perangkat keras yang diaktifkan oleh **KVM**.
- Dalam mode apa pun (baik sebagai virtualizer atau emulator), QEMU TIDAK HANYA mengemulasi prosesor, juga mengemulasi periferal yang berbeda, seperti disk, jaringan, VGA, PCI, port serial dan paralel, USB, dan sebagainya. Terlepas dari emulasi perangkat I / O ini, saat bekerja dengan KVM, QEMU-KVM membuat dan menginisialisasi mesin virtual.

Qemu

- QEMU mendukung **emulasi sistem penuh** di mana sistem operasi yang tanpa dimodifikasi dapat dijalankan dalam mesin virtual dan dalam **emulasi modus pengguna linux** dimana sebuah proses linux dikompilasi untuk sebuah target CPU yang dapat dijalankan di CPU lain.
- Download https://www.qemu.org/download/
- Latest release 5.1.0 , Aug 11th 2020

Penggunaan Qemu

- Penggunaan utama dari QEMU adalah untuk menjalankan satu sistem operasi yang lain, seperti Windows di Linux atau Linux di Windows.
- Penggunaan lain adalah debugging karena mesin virtual dapat dengan mudah dihentikan, dan status (state) -nya dapat diperiksa, disimpan dan dipulihkan.
- Selain itu, perangkat embedded tertentu dapat disimulasikan dengan menambahkan deskripsi mesin baru dan perangkat tiruan baru.

Qemu subsystem

- QEMU dibuat dari beberapa subsistem:
  - CPU emulator (currently x86 1 , PowerPC, ARM and Sparc)
  - Emulated devices (e.g. VGA display, 16450 serial port, PS/2 mouse and keyboard, IDE hard disk,NE2000 network card, …)
  - Generic devices (e.g. block devices, character devices, network devices) used to connect the emulated devices to the corresponding host devices

Qemu subsystem

- Machine descriptions (e.g. PC, PowerMac, Sun4m) instantiating the emulated devices
- Debugger
- User interface

Dynamic translator

- Penerjemah dinamis (Dynamic translator) melakukan konversi runtime dari instruksi intrsuksi CPU target menjadi sekumpulan instruksi instruksi host . Menghasilkan kode biner yang disimpan dalam cache translation sehingga dapat digunakan kembali.
- Keuntungan dibandingkan dengan sebuah interpreter adalah bahwa instruksi instruksi target diambil dan dikodekan hanya sekali saja.

Dynamic translators

- Biasanya dynamic translators sulit untuk mem-porting dari satu host to host yang lainnya karena keseluruhan generator kode harus ditulis ulang.
- Hal ini merepresentasikan tentang jumlah yang sama dari pekerjaan ketika menambahkan sebuah target ke kompiler C.
- QEMU sangat sederhana karenanya hanya merangkai potongan-potongan kode mesin yang dihasilkan secara off line oleh GNU C Compiler

Dukungan format disk image

- QEMU mendukung format disk image sbb:
  - OS X Universal Disk Image Format (.dmg) – Read-only
  - Bochs – Read-only
  - Linux cloop – Read-only
  - Parallels disk image (.hdd, .hds) – Read-only
  - QEMU copy-on-write (.qcow2, .qed, .qcow, .cow)
  - VirtualBox Virtual Disk Image (.vdi)
  - Virtual PC Virtual Hard Disk (.vhd)
  - VMware Virtual Machine Disk (.vmdk)
  - Raw images (.img) that contain sector-by-sector contents of a disk
  - CD/DVD images (.iso)

Network emulation

- QEMU dapat mensimulasikan beberapa kartu jaringan (PCI atau ISA kartu pada target PC) dan dapat menghubungkan mereka ke Virtual Local Area Network (VLAN).
- Perangkat host TAP dapat dihubungkan ke sejumlah QEMU VLAN. Ini adalah cara standar untuk menghubungkan QEMU ke jaringan real. QEMU menambahkan perangkat jaringan virtual pada host Anda (disebut tap), dan Anda kemudian dapat mengkonfigurasi seolah-olah itu adalah kartu ethernet nyata.

Instalasi software qemu

```
# sudo apt-get install qemu qemu-user
qemu-system qemu-utils bridge-utils
```

Membuat disk image

```
# qemu-img create -f qcow2 disk.img 256M
```

atau

```
# qemu-img create -f raw disk.img 256M
```

Emulasi sistem penuh (Instalasi OS)

```
# qemu-system-x86_64  -hda
mikrotik.img  -cdrom ./mikrotik-
6.46.8.iso  -boot d -m 256M
```

Emulasi sistem penuh (Running OS)

- Image qemu

  ```
  #qemu-system-x86_64  -hda  disk.img -netdev
  tap,id=tapnet,name=tap0 -device
  e1000,netdev=tapnet -m 256M
  ```

- Disk format virtualbox (.vdi)
- `# qemu-system-x86_64 -hda Ubuntu-14.04.vdi -netdev tap,id=tapnet,name=tap0 -device e1000,netdev=tapnet -m 256M`
- Disk format vmware (.vmdk)
- `# qemu-system-x86_64 -hda chr-6.35.vmdk -netdev tap,id=tapnet,name=tap0 -device e1000,netdev=tapnet  -m 256M`

Running OS dengan multiple interface network

- **Satu interface:**
- #qemu-system-x86_64 -hda **chr-6.47.7.vmdk** -netdev tap,id=mynet0,ifname=ether1,script=no,downscript=no -device e1000,netdev=mynet0 -m 256M
- **Dua interface:**

  #qemu-system-x86_64 -hda mikrotik.img -netdev tap,id=mynet0,ifname=ether1,script=no,downscript=no -device e1000,netdev=mynet0 -netdev tap,id=mynet1,ifname=ether2,script=no,downscript=no -device e1000,netdev=mynet1 -m 256M

Connecting to real network

```
# brctl addbr br0
# brctl addif br0 eth0
# ifconfig br0 10.10.10.1/24
# qemu-system-x86_64 -hda chr-6.35.vmdk   -netdev tap,id=mylan,ifname=tap0 -device e1000e,netdev=mylan -m 256M
# brctl addif br0 tap0
```