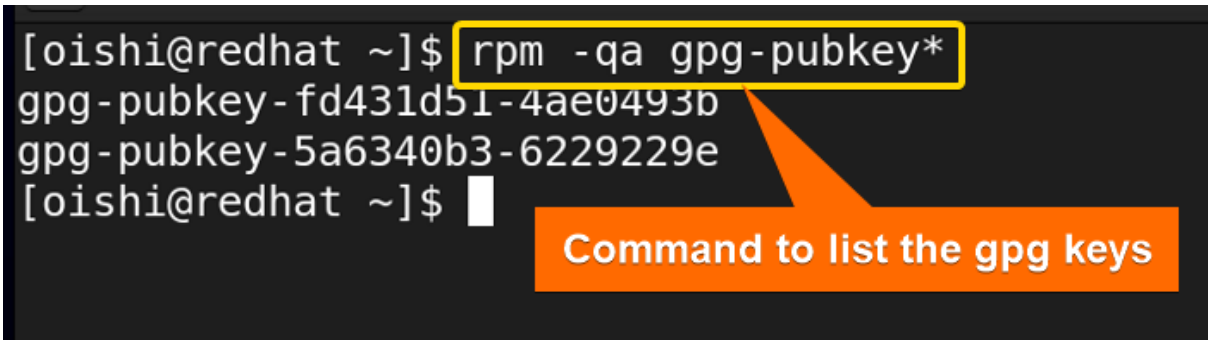
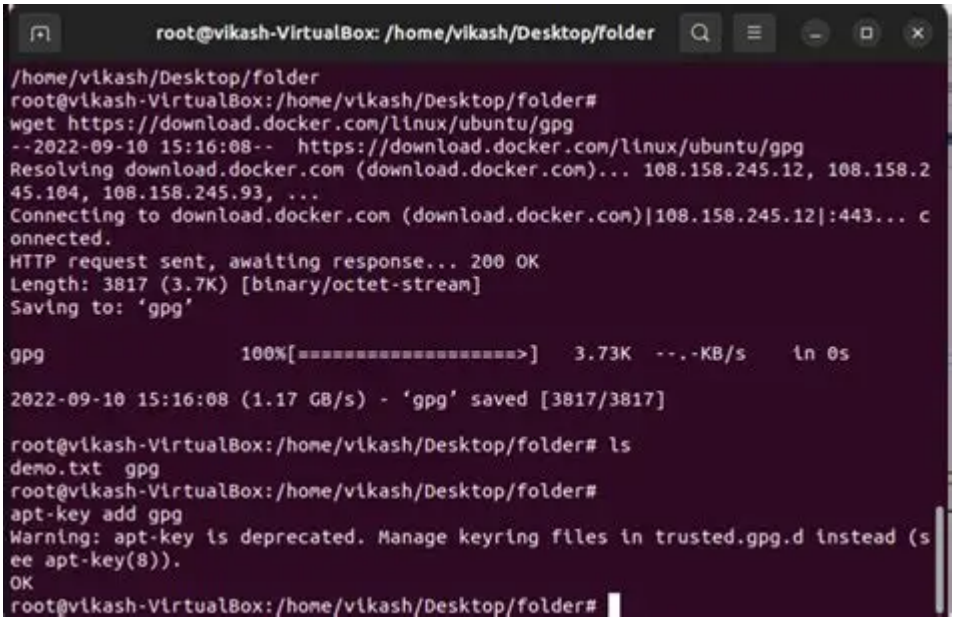
LPIC1 (T1)
Ghazaleh Keyvani
2025/7/7

XXXXXXXXXXXXXXXXXXXXXXXXXXXX

1. مکانیزم‌های امنیتی مخازن پکیج:

1. امضای دیجیتال GPG:
   - همه پکیج‌ها با کلید خصوصی مخزن امضا می‌شوند
   - سیستم کاربر کلید عمومی مخزن را دارد (ذخیره در `/etc/apt/trusted.gpg.d/`)
   - هنگام نصب، امضا بررسی می‌شود:
   - ```bash
     sudo apt update
     ```
   -


     ```



- `# بررسی خودکار امضاها توسط apt`

2. چکسام‌های امنیتی:
   - هر پکیج چکسام SHA256 دارد
   - لیست چکسام‌ها در فایل `Release` ذخیره می‌شود
   - این فایل هم با GPG امضا می‌شود
3. HTTPS و تصدیق هویت:
   - ارتباط امن با سرورهای مخزن
   - پیشگیری از حملات مرد میانی
4. سیستم اعتماد زنجیره‌ای:
   - پکیج‌ها از منابع معتبر زنجیره‌ای امضا دریافت می‌کنند
   - به‌روزرسانی خودکار کلیدهای امنیتی

---

2. اجزای مخزن APT در Ubuntu:

مثال	توضیح	مولفه
`focal main`	نرم‌افزارهای متن‌باز پشتیبانی شده توسط Canonical	main

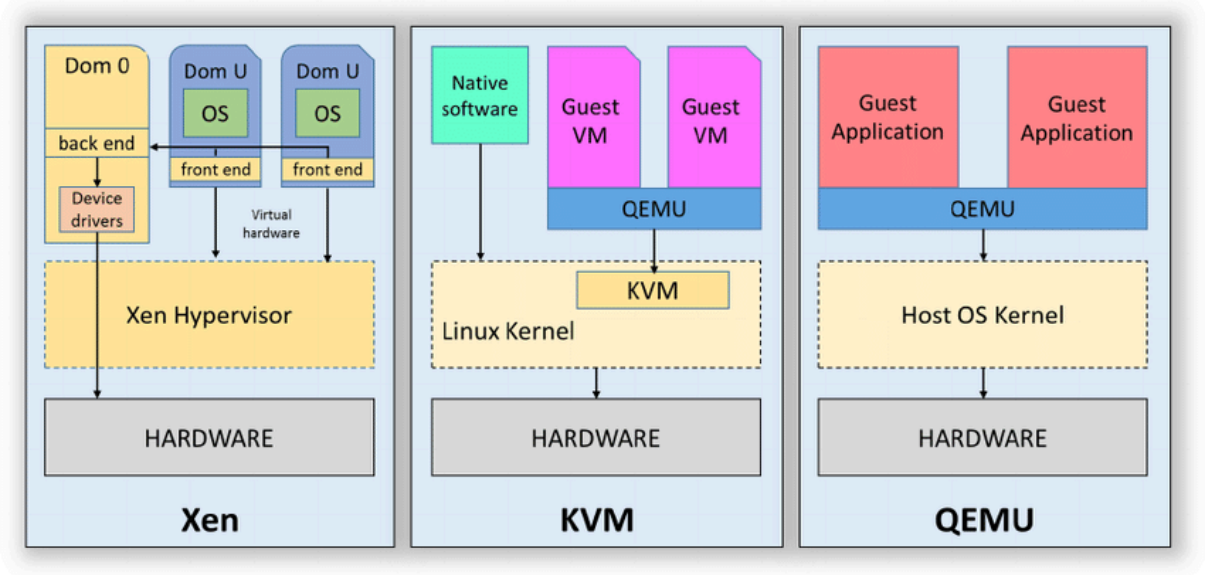
restricted	نرم‌افزارهای اختصاصی با پشتیبانی محدود	`focal restricted`
universe	نرم‌افزارهای متن‌باز پشتیبانی شده توسط جامعه	`focal universe`
multiverse	نرم‌افزارهای غیرآزاد با محدودیت‌های قانونی	`focal multiverse`
updates	به‌روزرسانی‌های مهم و رفع باگ	`focal-updates`
backports	نسخه‌های جدیدتر نرم‌افزارها برای نسخه قدیمی سیستم	`focal-backports`
security	وصله‌های امنیتی حیاتی	`focal-security`

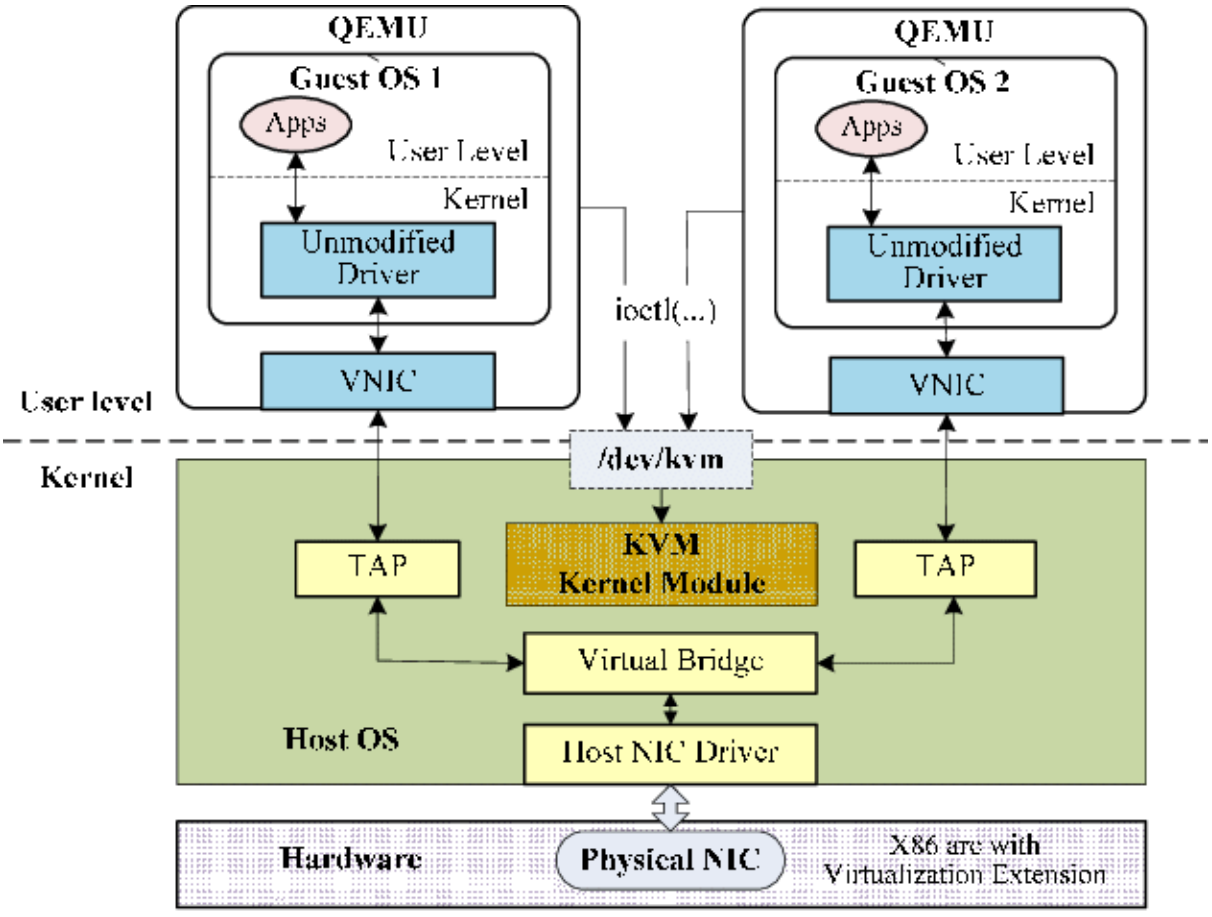
نکته: در مثال شما:

- خطوط `focal`: نسخه پایه 20.04
- خطوط `focal-updates`: به‌روزرسانی‌ها
- خطوط `focal-security`: اپدیت‌های امنیتی
- خط `focal-backports`: نسخه‌های جدیدتر اختیاری

---

3. تفاوت QEMU، KVM و qemu-kvm:





1. QEMU:
   - شبیه‌ساز سخت‌افزار کامل
   - قابلیت اجرای سیستم‌عامل‌های مختلف معماری‌ها (x86, ARM, ...)
   - بدون شتاب‌دهی سخت‌افزاری، کند است
2. KVM:
   - ماژول کرنل لینوکس برای مجازی‌سازی سخت‌افزاری

- از امکانات CPU (Intel VT-x/AMD-V) استفاده می‌کند
- فقط مدیریت منابع را انجام می‌دهد

3. qemu-kvm:
   - ترکیب QEMU و KVM
   - QEMU برای شبیه‌سازی دستگاه‌ها
   - KVM برای شتاب‌دهی مجازی‌سازی
   - نتیجه: مجازی‌سازی پرسرعت نزدیک به سخت‌افزار واقعی

پیاده‌سازی:

```bash
# بررسی پشتیبانی سخت‌افزار
grep -E '(vmx|svm)' /proc/cpuinfo

# نصب qemu-kvm
sudo apt install qemu-kvm libvirt-daemon-system
```

---

4. تفاوت VM و Container و کاربردها:

ویژگی	ماشین مجازی (VM)	کانتینر (Container)
سطح مجازی‌سازی	سخت‌افزار کامل	سطح سیستم‌عامل (کرنل مشترک)
سربار	سیستم‌عامل کامل VM بالا (هر دارد)	بسیار پایین

(فوری (میلی‌ثانیه	(کند (ثانیه تا دقیقه	بوت
نسبی (namespaces/cgroups)	کامل	ایزولیشن

توصیه‌ها:

دلیل	توصیه	سناریو
دسترسی مستقیم به دیسک، مدیریت بهتر I/O	VM	سنگین I/O دیتابیس با
سبک، گسترش آسان، مدیریت منابع کارآمد	Container	برنامه وب بدون حالت (Stateless)
تخصیص منابع اختصاصی و پایدار	VM	برنامه یکپارچه با نیاز منابع بالا
دسترسی مستقیم به سخت‌افزار	VM	برنامه نیازمند درایور سخت‌افزار خاص
پشتیبانی بهتر از درایورهای گرافیکی	VM	برنامه‌های گرافیکی (GUI)

محیط سیستم‌عامل کامل برای سرویس‌ها	VM	برنامه‌های وابسته به سرویس‌های بوت
نیاز به تغییر کرنل میزبان	VM	برنامه نیازمند ماژول‌های کرنل سفارشی
بهره‌وری منابع، اجرای سریع	Container	برنامه‌های سبک با حداقل نیاز منابع
اشتراک‌گذاری حافظه، ارتباط آسان	Container	برنامه‌های نیازمند تعامل مستقیم با پردازش‌ها

---

5. تفاوت الگوریتم Hash و Encryption:

Encryption (مثلاً AES)	**Hashing (مثلاً SHA-256)**	**ملاک**
تأمین محرمانگی داده‌ها	تأمین یکپارچگی داده‌ها	هدف

(قابل بازگشت با کلید (دوطرفه	غیرقابل بازگشت ((یک‌طرفه	بازگشت پذیری
طول متناسب با ورودی	طول ثابت (مثلاً 256 بیت)	خروجی
محافظت از داده‌های حساس	احراز هویت، تشخیص تغییرات	استفاده
`openssl enc -aes-256-cbc -in file.txt`	`sha256sum file.txt`	مثال عملی

---

6. تولید چکسام برای تشخیص تغییرات:

1. تولید چکسام‌ها:
2. bash
```
# تولید چکسام SHA-256 برای همه فایل‌ها
```
3. `find /path/to/config -type f -exec sha256sum {} \; > config_checksums.sha256`
4. بررسی تغییرات:
5. bash
```
# بررسی یکپارچگی فایل‌ها
sha256sum -c config_checksums.sha256

# خروجی نمونه:
file1.conf: OK #
```

```
file2.conf: FAILED #
```

6. `sha256sum: WARNING: 1 computed checksum did NOT match #`
7. پیاده‌سازی پیشرفته:
8. bash

```
# اسکریپت خودکار بررسی

bin/bash/!#

if ! sha256sum -c config_checksums.sha256; then

    echo "تغییرات غیرمجاز شناسایی شد!"

    # ارسال اعلان یا اقدامات امنیتی
```

9. `fi`

مزایا:

- تشخیص هرگونه تغییر حتی 1 بیت
- اجرای دوره‌ای با cron
- امکان ذخیره چکسام‌ها در مکان امن خارج از سرور