Securely booting confidential VMs with encrypted disk

Dov Murik <dovmurik@linux.ibm.com>

Tobin Feldman-Fitzthum <tobin@ibm.com>
Scope

• Confidential VMs
  • Untrusted host

• Approaches for encrypting guest’s disk

• Goal: start community discussions
Criteria for approaches

• Confidentiality: host can’t read passphrase or disk content
• Integrity: host can’t modify disk content
• Work with (all?) VM-based TEEs
• Simple to use: build image without TEE
  • For example: can be booted locally by entering the passphrase
Encrypted qcow2

- qcow2 supports encrypted disk luks
  - Also for remote storage (librbd / Ceph)
- VMM has disk passphrase and performs decryption
- Guest sees plaintext sectors

- Not suitable for confidential computing setting
  - Host (VMM) is untrusted
  - Cannot have disk passphrase
  - Cannot see plaintext disk data
Full disk encryption – 1


• Build a stripped-down grub:
  • Only needed modules
  • New efisecret module (not upstream)
  • Hard-coded grub.cfg that reads passphrase from SEV secret injection area

• Build a stripped-down OVMF (OvmfPkg/AmdSev)

• Embed grub.efi inside OVMF binary so it’s measured

• At launch, Guest Owner injects secret (disk passphrase); grub decrypts disk and continues boot
Full disk encryption – 2

Pros:
• Everything is measured up to the disk unlocking
• Kernel and initrd are inside the encrypted envelope

Cons:
• Relies on very early secret injection: doesn’t work as-is with SNP and TDX
• Grub patches are not upstream
• OVMF-with-embedded-grub complicates packaging and updates
cryptsetup in initrd – 1

- Similar to approach described in sev-guest repo https://github.com/AMDESE/sev-guest
- But - kernel+initrd+cmdline given with direct boot (⇒ measured using QEMU/OVMF hashes table)
- Disk has luks-encrypted root partition
- In init script:
  1. Send request (HTTP) to owner
  2. Retrieve nonce
  3. Request signed attestation report from SNP/TDX hardware
  4. Send report to owner (which verifies the report)
  5. Retrieve disk passphrase
  6. (In SEV: use efi_secret kernel module (>=5.19) to read the injected secret)
  7. Pass passphrase to cryptsetup luksOpen
cryptsetup in initrd – 2

• Measuring firmware is not enough: malicious kernel or initrd can steal the passphrase.

• SEV/-ES: Measured direct boot supported (QEMU, edk2)
  • KVM Forum 2021 talk: Securing Linux VM boot with AMD SEV measurement
    https://youtu.be/jzP8RlTRErk

• SNP: Similar approach (RFC patches posted)

• TDX: Should be solved by extending RTMRs
  • Direct boot (edk2 extends fw_cfg blobs); or
  • EFI partition (grub), unencrypted /boot partition (grub extends)

• Each TEE has its own way of attestation and secret injection
  • Maybe this can be abstracted by latchset/clevis (with pins)
cryptsetup in initrd – 3

Pros:
• Works on SEV/-ES, SNP, TDX

Cons:
• Kernel, initrd, cmdline are not encrypted and passed from host
  • OK if these components are immutable
  • Mix of encrypted root but unencrypted /boot might be dangerous
  • How does the owner update the kernel/initrd?
• Measurement must include kernel, initrd, cmdline – harder to verify
• Hardening is difficult
  • For example, can the host operator access emergency shell and steal the passphrase?
  • But this is true for the entire lifetime of confidential guest
Disk passphrase in vTPM – 1

• Physical machines with TPM can fetch FDE passphrase which was sealed to the TPM:
  • cryptsetup: *tpm2-initramfs-tool unseal*
  • grub: *cryptomount -k tpm2* (not upstream)
    [PATCH v2 0/5] Automatic TPM Disk Unlock (Hernan Gatta)

• Can we do the same for a confidential VM?
Disk passphrase in vTPM – 2

Pros:
• Everything is measured up to the disk unlocking
• Kernel and initrd are inside the encrypted envelope

Unknowns:
• Where is the vTPM running?
  • Can the host access its memory? Can the guest OS modify it?
  • SNP: Use VMPLs and leverage AMDESE/linux-svsm
• Does the vTPM have non-volatile storage?
  • Can the host read/modify it?
Confidential qcow2?

- Suggested-by: jejb
- Standard qcow2-encrypted image (local, or in Ceph)
- But VMM doesn’t decrypt; it passes encrypted sectors to guest
- Guest decrypts so OS sees a plaintext disk drive
  - ... and grub, and OVMF
- Q: What exactly in the guest is decrypting?
- Q: How is the passphrase injected securely into the guest?
- Q: Can we perform remote attestation + secret injection early enough in the guest’s life?
Thank You!

Securely booting confidential VMs with encrypted disk

Dov Murik <dovmurik@linux.ibm.com>

Tobin Feldman-Fitzthum <tobin@ibm.com>