CIQ

Secure Boot: Getting To Know Your Frenemy

Michael L. Young | Principal Systems Engineer

About

- Over 25+ years in the technology industry
- Member of Rocky Enterprise Software Foundation and Rocky Linux project
- Part of Fedora Community
- Over the years, contributor to projects such as Asterisk, Rocky Linux, Warewulf
- Maintain DisplayLink driver RPM packages
  (github.com/displaylink-rpm/displaylink-rpm)
- Work for CIQ as a Principal Systems Engineer
  - CIQ is the leading corporate sponsor of the Rocky Linux project
  - Focus on providing next generation infrastructure for modern high performant workloads

Secure Boot: Frenemy?

- Becoming a requirement in many environments
- Cloud VMs
- Differing opinions around if it solves a problem

What is Secure Boot?

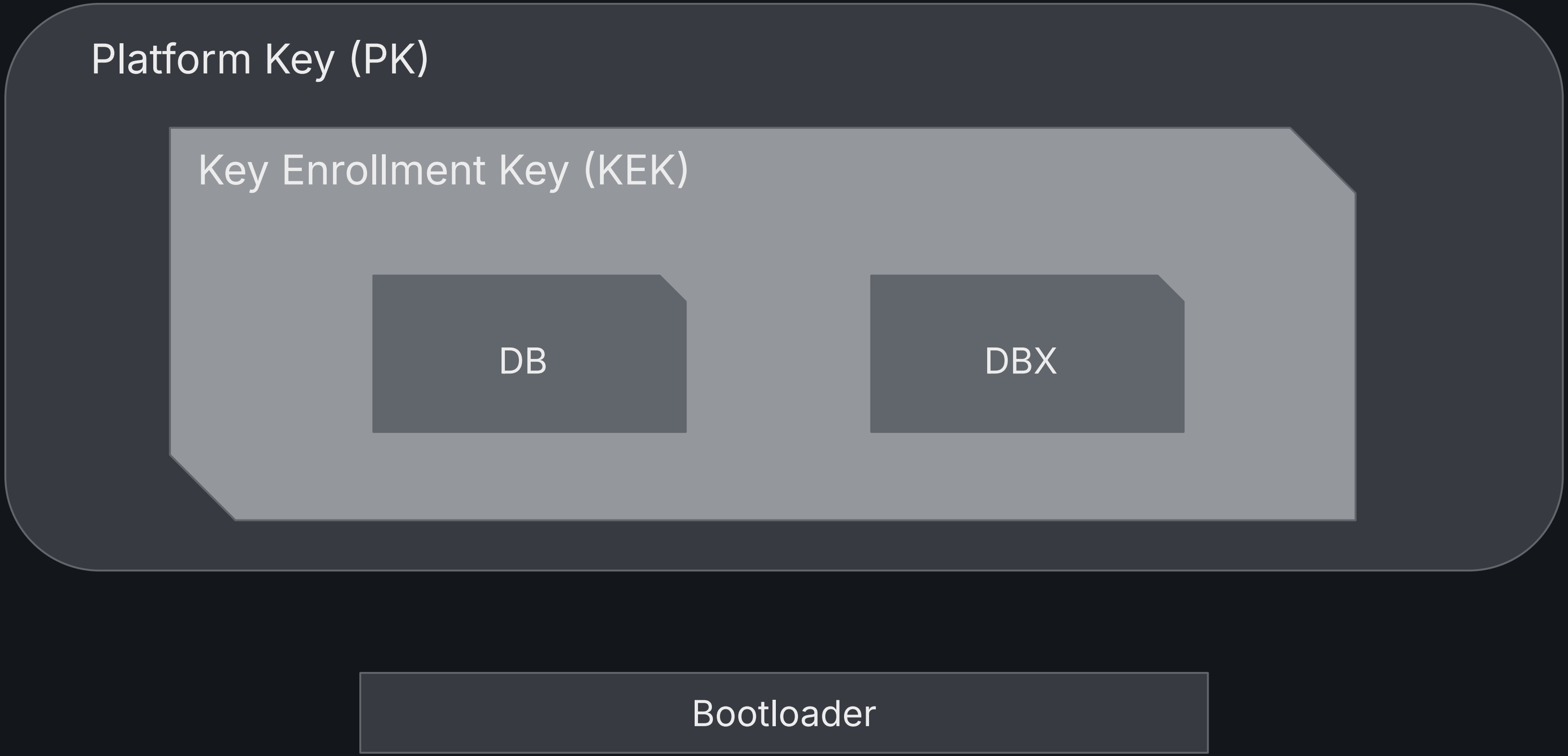
History

- 1970s - BIOS
  - Bootloader stored in MBR
  - Unable to validate trustworthiness
- 2000s - UEFI Specification
  - Remove limitations
  - Secure Boot added to spec 2006

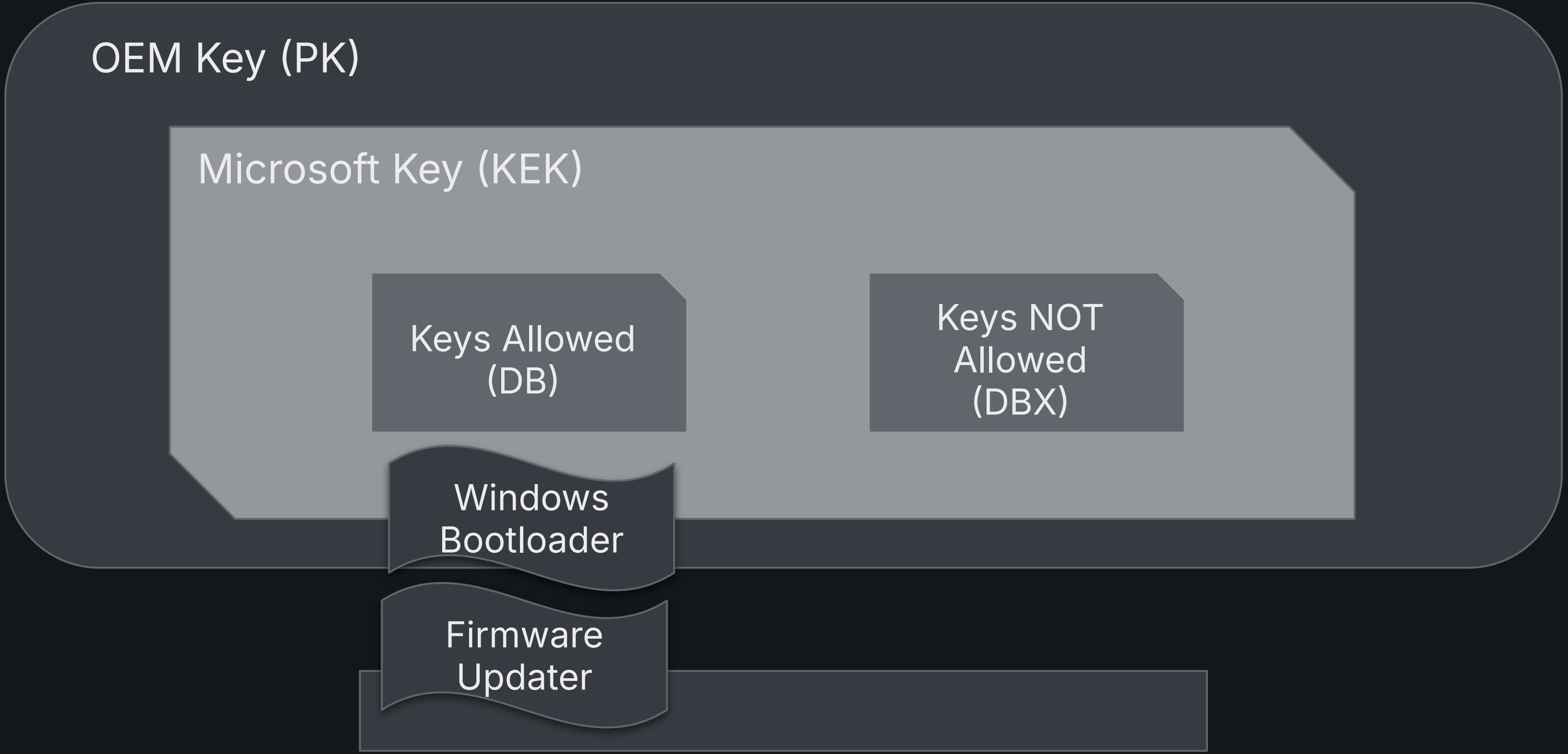
Goal

- Only allow trusted software to be run during boot
- Uses digital signatures for validation

How does it work?

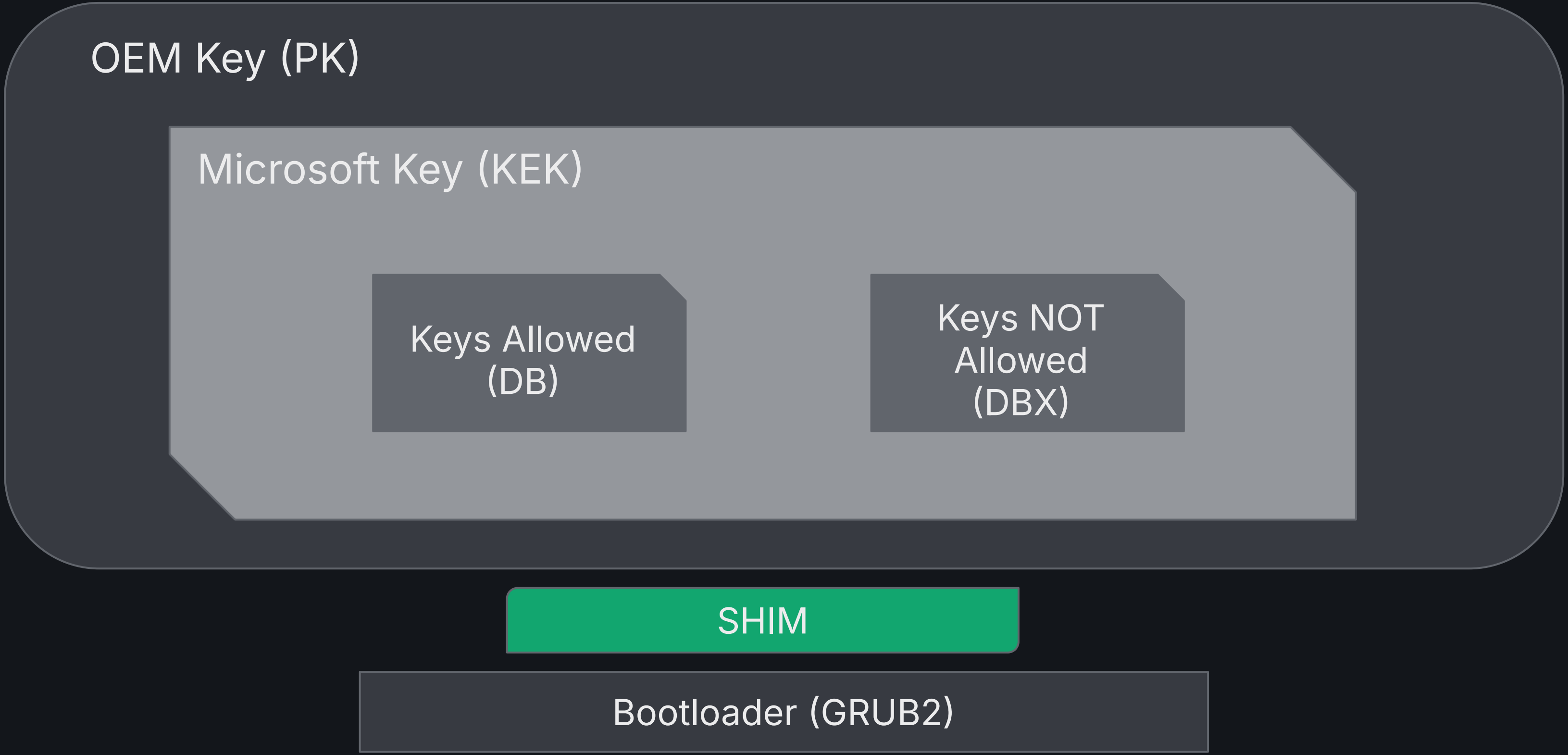


Simple Terms



CIQ

Linux


OEM Key (PK)
Microsoft Key (KEK)
Keys Allowed (DB)
Keys NOT Allowed (DBX)
SHIM
Bootloader (GRUB2)

How to get SHIM signed?

- Microsoft Hardware Program
    - Add EV code signing cert
- SHIM review
    - https://github.com/rhboot/shim-review
- Once SHIM is approved by the committee submit to Microsoft for signing
    - CAB file that contains your unsigned SHIM
- Receive back a signed SHIM

June 27, 2026

- The Microsoft keys for KEK and UEFI from 2011 expire
- What does this mean?
    - Your current SHIM will still work...
        - As long as your firmware is not updated with the new keys from 2023
    - New install media that contains a SHIM signed with the 2023 keys will not work if firmware has not been updated
- Microsoft has a site with more information and guidance from OEMs

CIQ

Out of tree kernel modules

- If not signed by a trusted CA in the database of keys, it will not be loaded
- Create your own signing key
- Enroll using MOK Manager
  - Machine Owner Key (MOK)
- Use this key for signing your kernel drivers

Gotchas

- Need to remember to manually re-sign module after kernel or driver updates

Out of tree modules

DKMS

- Can use same signing key that you created yourself and has been enrolled (MOK)
- On kernel updates, the module will be built against kernel being installed
- Automatically signed after being built

Attempt at a

DEMO

Trends

- Requirement in the Enterprise to keep Secure Boot enabled
- Customers in highly regulated industries requiring Secure Boot be enabled
- Cloud compute instances are adding supporting for Secure Boot
- Confidential computing
    - Hardware based Trusted Execution Environment (TEE)

CIQ

Takeaways

- Frenemy?
  - We may not like the current solution but we have to work with it.
  - It is becoming a requirement for many environments, including the cloud.
- Is there a better way?
  - Probably
- What is it?
  - We don’t know yet.
  - Needs to simplify management and not allow one entity to control it all

CIQ

Thank You!

Please stop by booth #412

myoung@ciq.com

@elgueromexicano.bsky.social

@elgueromexicano

/mlyoung

@elgueromexicano@fosstodon.org

ciq.com